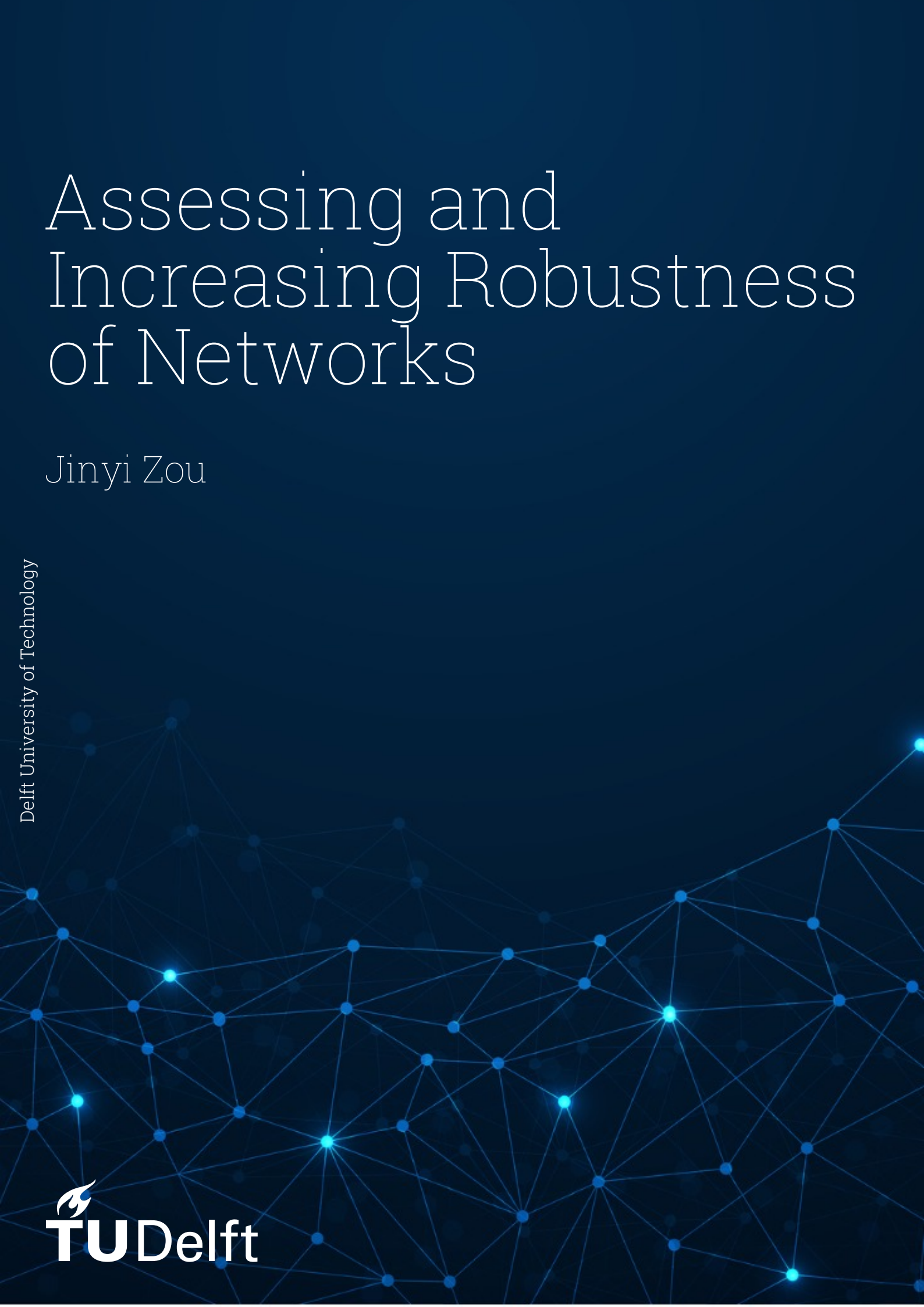
Assessing and Increasing Robustness of Networks

Jinyi Zou

Delft University of Technology



TU Delft

Assessing and Increasing Robustness of Networks

by

Jinyi Zou

to obtain the degree of Master of Science
in Electrical Engineering
Track Wireless Communication and Sensing
at the Delft University of Technology,
to be defended publicly on Monday December 4, 2023 at 14:00.

Student number:	5521424	
Project duration:	December 22, 2022 – December 4, 2023	
Thesis committee:	Prof. dr. ir. R.E. Kooij,	TU Delft, chair
	Dr. ir. Huijuan Wang ,	TU Delft
Supervisors:	Prof. dr. ir. R.E. Kooij,	TU Delft, supervisor
	Fenghua Wang,	TU Delft, daily supervisor

TU Delft

Preface

With this thesis titled 'Assessing and Increasing the Robustness of Networks', I fulfill the requirements for obtaining the Master of Science degree in Electrical Engineering from Delft University of Technology. The research was conducted under the guidance of my supervisor, Rob Kooij, who provided me with this valuable opportunity. Throughout the graduation period, he offered me constant encouragement and direction. Upon experiencing a state of indolence, his criticism helped me regain my motivation. I extend my gratitude to Fenghua Wang, my daily supervisor, whose valuable insights and guidance proved instrumental in the completion of this thesis. She patiently taught me how to utilize the cluster for program execution and assisted me in overcoming various challenges. Her encouragement always cheered me up when I encountered difficulties. I would also thank the team of Guanrong Chen. The team members provided me with guidance to the machine learning approaches implemented by them, offered me their dataset and codes, and patiently answered all of my questions. Finally, I would like to express my heartfelt gratitude to my parents. Their unwavering support and boundless love empowered me to persevere and carry on with my education in moments of adversity and setbacks.

Jinyi Zou
Delft, November 2023

Abstract

This thesis aims to assess robustness of networks by evaluating the performance of node attack strategies, the applicability and accuracy of different approaches, and to increase robustness of networks through analysing protecting methods, including link addition and node protection strategies. To be specific, the relative size of the Largest Connected Component (rLCC) and Average Two-Terminal Reliability (ATTR) are chosen to be the performance metrics to evaluate the robustness of networks. In the assessment of network robustness, simulations are employed for ten distinct attack strategies, which include random attack, non-updated and updated degree attacks, non-updated and updated stochastic degree attacks, non-updated and updated betweenness attacks, and greedy attack. Three analytical approximation methods are implemented and discussed, alongside the testing of a machine learning-based approach. Regarding the enhancement of network robustness, the comparative analysis involves thirteen link addition strategies and four node protection strategies. In all node removal scenarios, it is observed that the updated betweenness attack emerges as the most harmful strategy. Moreover, analytical approximations are proved to be effective means of evaluating network robustness in the scenario of predicting robustness for synthetic networks under random attack. Machine learning based methodologies show high accuracy in predicting robustness of synthetic networks, exhibiting acceptable error rates in predicting robustness of real-world networks. Besides, the updated betweenness link addition strategy and targeted betweenness-based node protection exhibit the highest efficacy in protecting networks against the most harmful attacks.

Contents

1 Introduction

Complex networks, such as those seen in social networks and the World Wide Web, provide a concise and potent means to represent the interconnected structure of diverse complex systems. Within these networks, nodes stand for entities, and edges signify various types of relationships or links [1][2]. In practical scenarios, numerous networks exhibit pronounced vulnerability to malicious attacks. Therefore, it is essential to develop effective methods for evaluating the robustness of existing networks in order to reduce potential losses. These assessments have a wide range of applications; for instance, consider the representation of a power grid system [3] as a network. This system is vulnerable to both intentional hostile actions and natural disasters, in addition to the effects of component aging and other contributing factors. The destruction of a power grid can result in widespread power outages, economic disruption, and public safety concerns, with long-term impacts on essential services and potential environmental consequences. In such a context, a comprehensive analysis of the consequences of various types of attacks becomes crucial in ensuring an uninterrupted and dependable power supply.

In this thesis, the performance of different attack strategies is evaluated according to the extent of reduction of the performance metrics rLCC and ATTR. The applicability and accuracy of different approaches assessing robustness of networks, including analytical approximations and machine learning-based approach, are analysed. Various increasing robustness methods are implemented and the performance of which is compared.

1.1. Objectives

The objectives of this thesis are:

1. Simulate how different attack strategies will impact on the robustness of different networks.

2. Try different approaches to assess the robustness of networks under specific attack strategies. Further investigate the performance of these methods.
3. Find approaches to increase the robustness of networks under attacks, and evaluate the performance of different approaches.

1.2. Related work

In this section, we review relevant literature on network robustness assessments and enhancement strategies, focusing on attack and protection mechanisms.

First of all, it is of significance to define what is robustness of networks. In [4], Van Mieghem *et al.* provide a description and a structured approach for assessing the robustness of network topology. It defines an 'R-value', which serves as a performance metric applicable to the service and is scaled to the range between 0 and 1. Consequently, an 'R-value' of 0 signifies the lack of network 'goodness', while an R-value of 1 indicates perfect 'goodness'. The relative size of the Largest Connected Component [5] and Average Two-Terminal Reliability [6] are two crucial metrics proposed to evaluate the robustness of networks, as they reflect the connectivity of a graph.

The influence of attacks on different networks is further investigated. Some researchers focus on node attack. Schneider *et al.* conduct simulations to evaluate the connectivity robustness [7] considering the size of the largest component during degree-based attack. In [8], Pu *et al.* analyze network controllability robustness in the presence of vulnerabilities across diverse network topologies. The investigation discerns that, in terms of its impact on network controllability robustness, degree-based attacks exhibit greater efficiency compared to random attacks. Furthermore, Lu *et al.* also conduct research on controllability robustness on networks in [9]. They take a deeper look in the controllability on both synthetic and real-world networks under attack strategies which are random, degree-based and betweenness-based.

To investigate changes in network robustness when nodes are removed, one can use the generating function derived from the degree distribution, see Newman *et al.* [10], who study the theory of random graphs with arbitrary distributions of vertex degree. They compute numerous statistical characteristics of random graphs through the utilization of generating functions. In [11], Kenett *et al.* evaluate the analytical structure and the results concerning percolation principles in a Network of Networks (NoN) consisting of n interconnected random networks. Based on the investigation, Kooij *et al.* further investigate the analytical approximations in [12], computing the minimum fraction of the number of driver nodes in directed networks subject to node removals.

For robustness prediction, there is a substantial amount of research using machine learning-based approaches to predict network robustness. For instance, a lot of work is done by Chen *et al.* [13, 14, 15, 16]. They conduct experiments based on machine learning approaches, using a knowledge-based Predictor for the Controllability Robustness (iPCR) [13], Learning Feature Representation-based Convolutional Neural

Network(LFR-CNN) [14] and Spatial Pyramid Pooling Convolutional Neural Network (SPP-CNN) [15] to predict the connectivity and controllability robustness of different networks. They compare the performance among different methods and further investigate the influence of the distribution of training data distributions [16].

In terms of increasing robustness, Masak *et al.* explore the effects of degree of nodes and the distance between the targeted nodes in link adding with two-step selections based on degrees or distances in [10]. In [17], Wang and Van Mieghem designed and compared two practical methods for stepwise maximizing the algebraic connectivity of a network that grows by adding edges. In [18], Louzada *et al.* propose a novel approach for altering the network's structure to enhance its robustness. The new rewiring method is based on the evolution of the network's largest component during a sequence of targeted attacks.

1.3. Contributions

The main contributions of this thesis are:

1. In simulations, the structural metrics rLCC and ATTR are used to evaluate the connectivity robustness of different networks.
2. Analytical approximations are computed on both real-world networks and synthetic networks. The performance of them is analysed.
3. The performance of a convolutional neural network model is evaluated.
4. Increasing robustness approaches, including link adding and node protection, are experimented with and explored in this thesis.
5. Metric energy is used to compare the performance of different attack and protecting strategies

1.4. Approaches

The step by step approaches used in this project are described below:

1. Carry out a literature study to learn existing approaches assessing network robustness and understand the current developments. Learn what kind of attack strategies can be applied in the simulation and what type of protecting strategies can be applied.
2. Learn how analytical approximation works to acquire a basic understanding of the algorithm. Study machine learning, learning the basic principals and how this approach can be applied in assessing the robustness of networks.
3. Apply different attack strategies on real-world networks (from Topology Zoo) of small size. Evaluate the performance of different attack strategies on networks with different degree sequences.

4. Apply different attack strategies on synthetic networks (Erdős-Rényi and Barabási-Albert networks). Evaluate the performance of different attack strategies on networks with a specific property.
5. Apply analytical approximations on both real-world and synthetic networks. Analyze the performance of this approach.
6. Train machine learning models with existing data, apply the models and compare the outcome with simulated results.
7. Apply different protection methods on both real-world and synthetic networks. Analyze the performance of these protecting methods.

1.5. Thesis outline

The structure of this thesis is as follows:

1. Chapter 2 introduces basic knowledge about graph theory, including graph metrics, performance metrics and graph models related to this thesis.
2. Chapter 3 describes different types of attack strategies, including random, targeted degree, targeted betweenness and greedy attacks, followed by a detailed elaboration of analytical approximations for random and targeted degree attacks. The Machine learning-based approach for predicting the robustness of the networks, called Spatial Pyramid Pooling Convolutional Neural Network (SPP-CNN), is also introduced.
3. Chapter 4 introduces two types of strategies for increasing the robustness of the networks, including link adding strategies and node protection strategies.
4. Chapter 5 represents the simulations, analytical approximations and predictions implemented in the experiments. The experiments encompass a variety of graph models, featuring both synthetic networks such as the Erdős–Rényi and Barabási–Albert graphs, as well as real-world networks primarily sourced from the Topology Zoo dataset. The performance of different attack strategies is compared. The performance of analytical approximations is discussed. The performance of the machine learning approach is explored. The performance of different increasing robustness strategies is compared and discussed.
5. Chapter 6 presents the conclusion and future work.

2

Background

In this chapter, related background is introduced. Section 2.1 introduces some basic knowledge about Node centrality, in which three types of node centrality used in the study are described. Section 2.2 introduces the performance metrics which represent the robustness of a network. Section 2.3 introduces the dataset, including synthetic networks and real-world networks, used in the experiments.

2.1. Node centrality

In graph theory, a graph is a fundamental mathematical structure [19] employed to represent relationships and connections between nodes and links. Node centrality indicates the topological importance of a node in a graph.

This section introduces three types of node centrality.

2.1.1. Degree

The degree of a node [19] is a fundamental concept that represents the number of neighboring nodes which are directly connected to the node. The degree of a node provides important information about the connectivity and structure of the network.

The degree of a node i is denoted as d_i in a graph $G(N, L)$, with N nodes and L links. The higher values of the degree of a node, the more important role the node plays in the network.

Some basic concepts concerning degree are mentioned as follows.

The maximum degree of a graph is defined as:

$$d_{\max} = \max_{i \in G} d_i. \tag{2.1}$$

The minimum degree of a graph is defined as:

$$d_{\min} = \min_{i \in G} d_i. \tag{2.2}$$

The average degree [19] of a graph, which can serve as an indicator to assess whether the graph exhibits a dense or sparse structure, is defined as:

$$E[D] = \frac{1}{N} \sum_{i=1}^{N} d_i = \frac{2L}{N}. \tag{2.3}$$

2.1.2. Betweenness

Betweenness [19] is a concept in network analysis that quantifies the importance of a node within a graph based on its role in facilitating shortest paths between pairs of other nodes. It measures how often a node lies on the shortest path between other nodes in the graph. Mathematically, the betweenness (b_i) of a node i in a graph G is defined as:

$$b_i = \sum_{s \neq v \neq t} \frac{\sigma_{st(i)}}{\sigma_{st}}, \tag{2.4}$$

where s and t are distinct nodes in the graph G, σ_{st} is the total number of shortest paths from node s to node t, and $\sigma_{st(i)}$ is the number of those shortest paths that pass through node i.

In simpler terms, betweenness measures the fraction of all shortest paths in the graph that pass through a particular node. Nodes with high betweenness are often critical for maintaining efficient communication and flow in networks. Their removals can have a significant impact on network connectivity and efficiency.

Some basic concepts concerning betweenness are mentioned in the following sections. The maximum betweenness of a graph is defined as:

$$b_{\max} = \max_{i \in G} b_i. \tag{2.5}$$

The minimum betweenness of a graph is defined as:

$$b_{\min} = \min_{i \in G} b_i. \tag{2.6}$$

2.1.3. Eigenvector

In the context of graph theory, the eigenvector centrality of a graph [20] refers to a numerical value associated with a specific mathematical representation of the graph, known as its adjacency matrix. The eigenvector centrality provides important

information about the structural properties of the graph and has various applications in network analysis, physics, and other fields.

The eigenvector centrality of a graph can be formally defined as follows:

Given a graph G with an adjacency matrix A, the centrality for node i is the $i-th$ element of an eigenvector associated with the largest eigenvalue λ of A. The eigenvector x is defined by the equation:

$$Ax = \lambda x. \tag{2.7}$$

Eigenvector centrality is significant because it provides insights into various graph properties, including the graph's connectivity, spectral gap, and other structural characteristics.

Some basic concepts concerning eigenvector centrality are mentioned in the following sections. The maximum eigenvector centrality of a graph is defined as:

$$x_{\max} = \max_{i \in G} x_i, \tag{2.8}$$

and the minimum eigenvector centrality of a graph is defined as:

$$x_{\min} = \min_{i \in G} x_i, \tag{2.9}$$

where x_i indicates the eigenvector centrality for node i.

2.2. Performance Metrics

Performance metrics [21] are a well-known area in the conventional analysis of graphs. They are always used to explain robustness of a network, and determine how much damage is induced by node or link removal. In this thesis, the robustness of a network is measured by two structural metrics, the relative size of the Largest Connected Component (rLCC) [5] and Average Two-Terminal Reliability (ATTR) [6]. The performance of different attack and protection strategies is measured by Energy. The detailed introductions to these Performance metrics are presented in the following sections.

2.2.1. rLCC

The rLCC is the ratio of the size of the largest cluster of connected nodes to the original number of nodes N [5]. It reflects the connectivity of the graph. Thus rLCC satisfies:

$$rLCC = \frac{LCC}{N}, \tag{2.10}$$

where LCC is the number of nodes in the largest connected component and N is the number of nodes in the considered network.

2.2.2. ATTR

ATTR is defined as the number of connected node pairs divided by the total amount of node pairs [6]. It corresponds to the probability that a randomly chosen pair of nodes can reach each other, and therefore provides a global measure of connectivity between all pairs of nodes. When the network is fully connected, ATTR equals one. In cases where the network is not fully connected, ATTR is determined by dividing the sum of the number of pairs of nodes in each connected component by the total number of pairs of nodes in the entire network. Therefore ATTR satisfies:

$$ATTR = \frac{\sum_{i=1}^{m} \binom{n_i}{2}}{\binom{N}{2}}, \tag{2.11}$$

where m is the number of connected components, n_i is the number of nodes in each connected component, and N is the number of nodes in the network.

Energy

Energy is used to evaluate the performance of different attacks and increasing robustness strategies. It is computed by averaging the summed up metrics (rLCC and ATTR in this thesis) throughout the attack process. The higher the energy is, the worse the performance of the attack strategy is, or the better the performance of the increasing robustness strategy is. It is defined by:

$$E_{metric} = \frac{\sum_{i=1}^{n_m} m_i}{N}, \tag{2.12}$$

where n_m denotes the number of removed nodes, m_i represents the metrics (rLCC or ATTR) after i node(s) is/are removed, and N is the number of nodes in the network.

2.3. Dataset

This section introduces the dataset used in the study, including three synthetic graphs and the real-world network dataset.

2.3.1. Synthetic graphs

A random graph is a mathematical model used in graph theory to represent a type of graph that is generated in a probabilistic or stochastic manner. In a random graph, the edges between nodes are determined based on a certain probability distribution or random process. In this thesis, three graph models are used.

Erdős–Rényi model

The Erdős–Rényi (ER) random graph model [22], named after mathematicians Paul Erdős and Alfréd Rényi, is a foundational model in network theory and graph theory.

It is employed to stochastically generate random graphs characterized by two primary parameters:

n: The total number of nodes in the graph.

p: The edge formation probability between any pair of nodes.

In the $G(n,p)$ model, a graph is created by connecting labeled nodes in a random manner. Each edge's inclusion in the graph is determined independently, with a probability of p. The degree distribution of the graph follows a binomial form, denoted as:

$$P(D=k)=\binom{n-1}{k}p^k(1-p)^{n-1-k}. \tag{2.13}$$

The average degree is:

$$E[D]=(n-1)p. \tag{2.14}$$

The ER model facilitates the generation of graphs exhibiting a spectrum of sparsity or density, contingent upon the p value, with lower p values yielding sparser graphs and higher p values resulting in denser graphs.

Barabási–Albert model

The Barabási–Albert (BA) model [23], introduced by Albert-László Barabási and Réka Albert, represents a stochastic graph model designed to emulate scale-free networks—a characteristic feature of various real-world networks. The BA model is characterized by two primary parameters:

n: The total number of nodes in the graph.

m: The number of new links established when adding a new node.

In the $G(n,m)$ model, a graph is created by adding links and nodes on a star network of $m+1$ nodes. To be specific, new nodes are subsequently introduced into the evolving graph, each connecting to m existing nodes. The selection of existing nodes to which a new node attaches is governed by a preferential attachment mechanism, with nodes more highly connected (i.e., possessing a higher degree) being more likely to attract new connections. With a node i of degree d_i, the probability of connection between a new node and the node i is denoted as:

$$p_i=\frac{d_i}{\sum_{j\in G}d_j}. \tag{2.15}$$

The BA random model is frequently deployed for the simulation of networks that mirror the structural properties encountered in diverse real-world network systems, rendering it a valuable instrument in the analysis of complex systems.

Configuration model

The Configuration Model [24] is a random graph model used in network science to generate graphs with a specified degree sequence. It can generate graphs with a prescribed degree distribution. The resulting graphs are typically random and may not capture other structural properties seen in real-world networks, such as community structure or clustering.

2.3.2. Topology Zoo

The Topology Zoo [25] is a collection of 233 interconnected and undirected network datasets, compiled from publicly available information provided by network operators. Many studies focused on evaluating and enhancing network robustness rely on data sourced from the Topology Zoo. In this thesis, the analysis of real-world networks is mainly based on networks from the Topology Zoo. Four networks are chosen to display the performance of different approaches as examples. Some basic properties are shown in the Table. 2.1

	$\|V\|$	$\|E\|$	D_{avg}
$Geant2012$	40	61	3.05
$Garr201103$	61	89	2.92
$Deltacom$	113	183	3.24
$UsCarrier$	158	189	2.39

Table 2.1: Basic properties of chosen networks in Topology Zoo. In the table, $|V|$ indicates the number of nodes; $|E|$ indicates the number of edges; D_{avg} indicates the average degree.

2.3.3. Network Repository

The Network Repository [26] stands as a pioneering resource, offering both interactive data access and network-related datasets for real-time visual analytics. It distinguishes itself not only as the first of its kind but also as the most extensive repository of network data, encompassing thousands of datasets across more than 30 domains, ranging from biological to social networks. This diverse and comprehensive collection of network graph data holds immense value, facilitating significant research discoveries and serving as a benchmark for various applications and fields, such as network science, bioinformatics, machine learning, data mining, physics, and social science. The repository includes a wide array of network data types, including relational, attributed, heterogeneous, streaming, spatial, and time series data, as well as non-relational machine learning datasets. All graph datasets are made easily accessible in a standardized format. Furthermore, the Network Repository boasts an interactive graph analytics engine that enables users to visualize network structures, glean macro-level statistics about the graph data, and explore essential micro-level properties of nodes and edges. Several networks are chosen to display the performance of different approaches as examples. Some basic properties are shown in Table. 2.2

	$\|V\|$	$\|E\|$	D_{avg}
$odepa400$	400	802	4.01
$power-494-bus$	494	1080	4.37
$netz4504-dual$	615	1171	3.81
$power-662-bus$	662	1568	4.74

Table 2.2: Basic properties of chosen networks in Network Repository. In the table, $|V|$ indicates the number of nodes; $|E|$ indicates the number of edges; D_{avg} indicates the average degree.

3

Assessing the robustness of networks by attacking nodes

To assess the robustness of networks, different attack strategies are applied and corresponding simulations are conducted. In the experiments, various types of attacks are considered that lead to node removals. Removals of links are not considered in this thesis. The detailed description of these attack strategies is shown in the Section 3.1. Afterwards, the prediction methods, including analytical approaches introduced and a Machine learning-based approach, are described in 3.2

3.1. Attack strategies

In graph theory, attack strategies involve different techniques for disturbing a network by singling out particular nodes within the graph. These methods are typically explored in the context of network robustness, aiming to assess how effectively a network can endure attacks. In this section, four types of attack strategies are introduced.

3.1.1. Random attack

In the random attack strategy, nodes are removed randomly, and the structural metrics are derived each time after a node is removed. To be specific, in this attack strategy, every node has the same probability to be removed, which equals $Pr(node) = \frac{1}{n}$, where n is the number of nodes in the network.

3.1.2. Targeted degree-based attacks

In the attack strategy based on degree, nodes are removed in regard to the degree of the node. In this section, two normal ways and two stochastic ways to remove nodes

are discussed.

Degree attacks

For degree attacks, nodes are removed directly based on the degree of the node. The types of removing are classified as non-updated and updated degree attacks, and they are introduced as follows:

- **Non-updated degree attack:** In non-updated degree attacks, the degree of each node in the network is derived before the attack is applied. Each node is removed in descending order according to its degree based on the originally derived degree sequence. In that process, if there are multiple nodes with the same value of degree, they will share the same probability to be removed and the algorithm will randomly pick one of them to be removed.
- **Updated degree attack:** In updated degree attacks, the degree of each node in the network is derived each time before an attack is applied, and then the node with the highest degree is picked to be removed. In that process, if there are multiple nodes with the same value of degree, they will share the same probability to be removed and the algorithm will randomly pick one of them to be removed.

Stochastic degree attacks

For the Stochastic degree attack, the probability of attacking a node, is proportional to some power of its degree. To be specific, if it is assumed that the probability of removing node i with degree k_i is denoted by p_i, then p_i satisfies:

$$p_i = \frac{k_i^\alpha}{\sum_{j \in N} k_j^\alpha} \tag{3.1}$$

where N indicates the node set in the network applied. Here α is a predefined parameter, indicating the extent to which the degree affects the node removal. The larger the absolute value of α is, the more the attack strategy is influenced by degree. To describe the effect of α in detail, if $\alpha = 0$, each node shares the same probability to be removed. It is regarded as random attack in that case. If $\alpha > 0$, the nodes with larger degree share higher probability to be removed. If $\alpha < 0$, the nodes with larger degree share lower probability to be removed. In the thesis, two cases are considered: $\alpha = 1$ and $\alpha = 10$.

- **Non-updated stochastic degree attack:** In non-updated stochastic degree attacks, the probability of each node to be removed is derived before the attack is applied. Each node is then removed according to the derived probability sequence.
- **Updated stochastic degree attack:** In updated stochastic degree attacks, the probability of each node to be removed is derived and renewed each time before a node is removed. The removal each time for each node is based on the updated derived probability sequence.

3.1.3. Targeted betweenness-based attacks

In targeted betweenness-based attacks, a node is removed based on the betweenness of the node. A node is removed directly based on the betweenness of the node. In this section, the types of removing are classified as non-updated and updated betweenness attacks. They are introduced in the following sections.

- **Non-updated betweenness attacks:** In non-updated betweenness attacks, the betweenness of each node in the network is derived before the attack is applied. Nodes are removed in descending order according to the originally derived betweenness sequence. In the process, if there are multiple nodes with the same value of betweenness, they will share the same probability to be removed and the algorithm will randomly pick one of them to be removed.
- **Updated betweenness attacks:** In updated betweenness attacks, the betweenness of each node in the network is derived each time before the attack applied, and then the node with the highest betweenness is picked to be removed. In the process, if there are multiple nodes with the same value of betweenness, they will share the same probability to be removed and the algorithm will randomly pick one of them to be removed.

3.1.4. Greedy attacks

For greedy attacks, the aim is to remove the node which will result in the smallest value of a metric in each step. In this thesis, the metric is chosen as rLCC. In that process, if there are multiple nodes which will result in the same smallest value of rLCC after they are removed, they will share the same probability to be removed and the algorithm will randomly pick one of them be removed.

3.2. Prediction strategies

Prediction methods in graph theory encompass both analytical approximations and machine learning-based approaches to forecast various characteristics or behaviors within networks. In this section, analytical approximations for rLCC and ATTR are demonstrated in detail at first, followed by the description of the machine learning-based approach.

3.2.1. Analytical Approximations

Analytical approximation algorithms [27] are efficient algorithms that find approximate values of structural metrics under different attacks. Conducting this study holds both practical value and significance due to its efficiency compared to simulations, offering a considerably shorter time requirement, while still yielding relatively precise results. This thesis discusses three analytical approximation methods, approximating rLCC

and ATTR under random attacks and stochastic degree attacks in two cases.

Analytical Approximations of rLCC under random attacks

To derive the analytical approximation of rLCC [10], the generating function [28] for the degree distribution of the network need to be firstly derived:

$$G_0(x) = \sum_{k=0}^{\infty} p_k x^k, \tag{3.2}$$

where $p_k = \frac{n_k}{N}$ (n_k denotes the number of nodes with degree k, and N denotes the total number of nodes), x is an arbitrary variable, and k is the corresponding degree. The average degree is denoted by

$$< k >= G_0'(1). \tag{3.3}$$

Afterwards, the generating function for excess degree distribution [29] is derived as:

$$G_1(x) = \frac{1}{< k >} \sum_{k=1}^{\infty} k p_k x^{k-1} = \frac{1}{< k >} G_0'(x). \tag{3.4}$$

Under random attacks, as the proportion of nodes being removed increases with the ongoing attack, the generating function for degree [30] becomes:

$$\overline{G_0}(x) = G_0(p + (1-p)x), \tag{3.5}$$

and the average degree is derived as:

$$\overline{< k >} =< k > (1-p). \tag{3.6}$$

Then the generating function for excess degree distribution is derived as:

$$\overline{G_1}(x) = \frac{1}{\overline{< k >}} \overline{G_0'}(x). \tag{3.7}$$

The relative size of the largest connected component S can be computed with the generating functions [29] for degree and excess degree distributions known. It is derived as:

$$S = 1 - \overline{G_0}(u), \tag{3.8}$$

where u is the smallest non-negative real solution of

$$u = \overline{G_1}(u). \tag{3.9}$$

Analytical Approximations of ATTR for random attack

The analytical approximation of ATTR is derived after the analytical approximation of rLCC is computed. The first step is to derive lower and upper bounds for the ATTR. Here, a network with N nodes and the size of the largest connected component LCC is considered. For the lower bound, it is assumed that all components outside the largest one are isolated nodes with degree 0. For this case, we obtain:

$$ATTR_{min} = \frac{\binom{LCC}{2}}{\binom{N}{2}}. \tag{3.10}$$

Under this assumption, the average size of the connected components other than the LCC, denoted by μ_{min}, satisfies:

$$\mu_{min} = 1 \tag{3.11}$$

For the upper bound, it is assumed that the remaining connected components are as large as possible. To determine the upper bound, a number H is defined as:

$$H = min\{LCC, N - LCC\}, \tag{3.12}$$

and two parameters Q and R are defined by the equation:

$$N - LCC = QH + R, \tag{3.13}$$

where Q indicates the number of second largest connected components, and R indicates the number of nodes in the smallest connected component in that case. The upper bound is then derived as:

$$ATTR_{max} = \frac{\binom{LCC}{2} + Q\binom{H}{2} + \binom{R}{2}}{\binom{N}{2}}. \tag{3.14}$$

In this case, the average size of the connected components outside the LCC (μ_{max}) is derived as follows: If $H = N - LCC$, it is derived as:

$$\mu_{max} = H, \tag{3.15}$$

if $H \neq N - LCC$ and $R = 0$, it is derived as:

$$\mu_{max} = \frac{N - LCC}{Q}, \tag{3.16}$$

if $H \neq N - LCC$ and $R \neq 0$, it is derived as:

$$\mu_{max} = \frac{N - LCC}{Q + 1}. \tag{3.17}$$

For the general case, according to [10], the average size of connected components outside the LCC, denoted as $\bar{X}$, is derived as:

$$\bar{X} = \frac{2}{2 - \overline{< k >} u^2/(1-S)}, \tag{3.18}$$

where S is the previously computed analytical approximation of rLCC and u is the solution of Eq.(3.9). With the average sizes of the connected components outside the LCC of the lower bound case and the upper bound case being derived, and the average size of connected components outside the LCC being known, a parameter β can be derived in such a way that the weighted average of μ_{min} and μ_{max} is equal to $\bar{X}$:

$$\bar{X} = \beta\mu_{min} + (1-\beta)\mu_{max}. \tag{3.19}$$

The parameter β is then solved as:

$$\beta = \frac{\bar{X} - \mu_{max}}{\mu_{min} - \mu_{max}}. \tag{3.20}$$

The final estimate for the ATTR is derived by taking a weighted average of $ATTR_{min}$ and $ATTR_{max}$ as:

$$ATTR^* = \beta ATTR_{min} + (1-\beta)ATTR_{max}. \tag{3.21}$$

Analytical Approximations for stochastic degree attacks

In the previous section, it is introduced that a parameter α is required to be fixed in advance, and we choose two cases with $\alpha = 1$ and $\alpha = 10$. The analytical approximations also focus on the corresponding two cases, approximating the structural metrics of networks under non-updated stochastic degree attacks. The derivation of the analytical approximations for non-updated stochastic degree attacks refers to the previous analysis of analytical approximations for random attacks. It is supposed that the changes of the generating functions for degree and excess degree distributions correspond to those in a random attack. In that case, the fraction p of removed nodes under targeted attacks are mapped onto the effective proportion $\overline{p}$ of nodes under random node attack. The aim is to derive the effective proportion $\overline{p}$ to substitute the original p mentioned in the previous section for computing the analytical approximations of rLCC under random attacks. With the $\overline{p}$ computed, rLCC and ATTR can be then computed following the steps described in the previous section.

Case of $\alpha = 1$

The derivation of $\overline{p}$ in that case is proposed in [12]. It is calculated by:

$$\overline{p} = 1 - \frac{f G_\alpha'(f)}{< k >}, \tag{3.22}$$

where $G_\alpha(x) = \sum_{k=0}^{\infty} p_k x^{k^\alpha}$, and in this case, $G_\alpha(x) = \sum_{k=0}^{\infty} p_k x^k$ with $\alpha = 1$ (where $p_k = \frac{n_k}{N}$, n_k denotes the number of nodes with degree k, and N denotes the total number of nodes). Here $< k >$ is the average total degree of the initial network. f is derived by the equation: $G_\alpha(f) = 1 - f$.

Case of $\alpha = 10$

The derivation of $\overline{p}$ in that case is proposed in [12]. It is calculated by:

$$\overline{p} = \frac{\sum_{k=k_{max}}^{k=\overline{k}} p_k k}{< k >}, \tag{3.23}$$

where k_{max} denotes the largest degree, and degree $\overline{k}$ is derived by the equation when $\overline{k}$ satisfies:

$$\sum_{k=k_{max}}^{k=\overline{k}} p_k = p. \tag{3.24}$$

3.2.2. Machine learning-based approach

A type of Convolutional Neural Network is developed by the team of Guanrong Chen, which is called the Spatial Pyramid Pooling Convolutional Neural Network (SPP-CNN)[15]. The novel framework introduces a spatial pyramid pooling layer positioned between the convolutional and fully-connected layers. This addresses the prevalent problem of incongruity found in CNN-based prediction methods, thereby enhancing its adaptability and extending its applicability.

As depicted in Figure 3.1, an $N \times N$ input image yields L $N' \times N'$ feature maps, where L denotes the quantity of filters in the last convolutional layer. Within the SPP layer, these feature maps are partitioned into three distinct tiers of spatial bins, sized at 1×1, 2×2, and 4×4, and subsequently subjected to max pooling with corresponding dimensions. Following this process, an output representation vector of size pL is produced as the SPP layer's output, where both L and p are predefined hyperparameters. Consequently, regardless of the input image's size, a fixed-length pL-vector is generated for input into the fully-connected layers. In the thesis, three pyramid pooling levels are employed, with sizes of 1×1, 2×2, and 4×4, respectively. Empirical evidence has confirmed that the performance of the SPP layer remains unaffected by variations in pyramid bin settings.

Prediction error

To evaluate the performance of the machine learning model, prediction error [31] is often used. In this thesis, the errors in prediction refer to the variances between the actual values of the dependent variable and the values predicted by the machine learning model. To be specific, we define $\mathbf{v_t} = \{v_t(i)\}_{i=0}^{N-1}$ and $\mathbf{v_p} = \{v_p(i)\}_{i=0}^{N-1}$ as the simulated and the predicted robustness curves respectively, where i indicates the number of experiments conducted. The prediction error ξ is defined as:

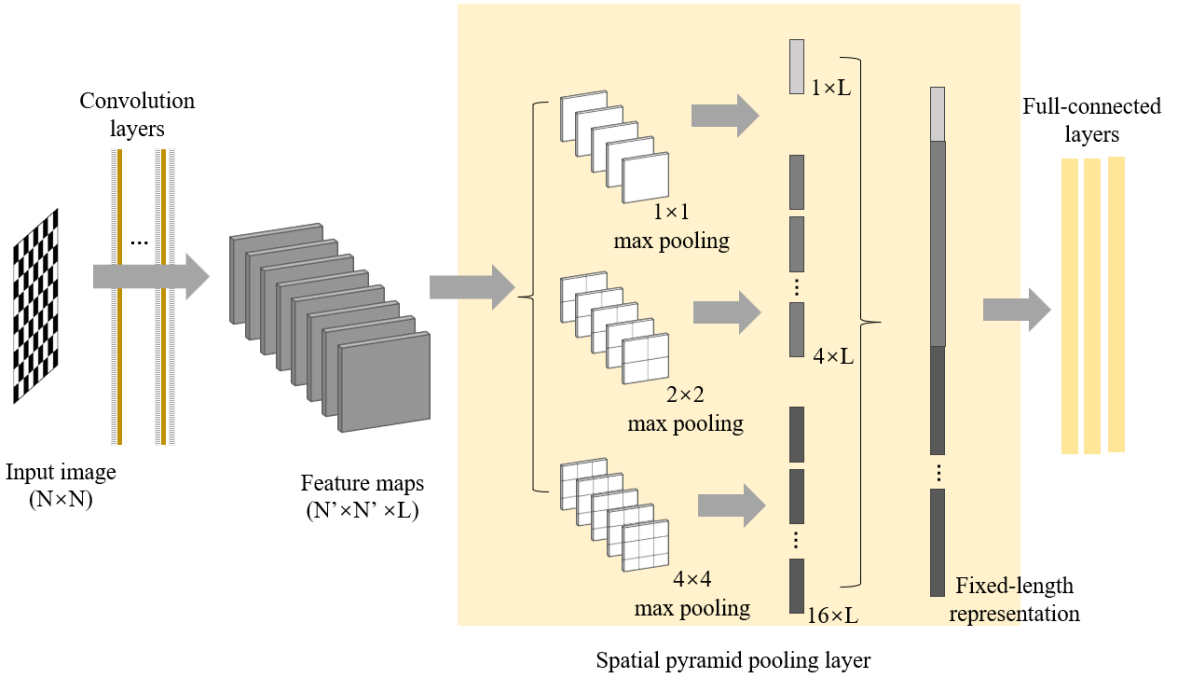


Figure 3.1: The convolutional neural network structure with a spatial pyramid pooling layer

$$\xi = \frac{1}{N}\sum_{i=0}^{N-1}\xi(i), \tag{3.25}$$

where $\xi(i) = |v_t(i) - v_p(i)|$, $i = 0, 1, \ldots, N-1$. The lower the value of the prediction error, the better the prediction performance of the model.

4

Increasing the robustness of networks

This chapter delves into the realm of network robustness, exploring approaches aimed at augmenting the robustness of networks. Two strategies for increasing robustness are investigated. Section 4.1 introduces a strategy achieved through the incorporation of additional links. Section 4.2 introduces a strategy realised by protecting nodes.

4.1. Link addition strategies

This section describes different link addition strategies in detail. Link addition strategies protect the networks by increasing their connectivity before the attacks are processed. To be specific, a specific number of links are added through specific approaches before attacks.

4.1.1. Random link additions

In random link addition strategy, links between each node are added randomly. To explain it in detail, in this strategy, every link in the complement graph of the graph is of the same probability to be added, which is $\frac{1}{n_c}$, where n_c is the number of edges in the complement graph of the network.

4.1.2. Targeted degree-based link additions

To explore the influence of degree, the targeted degree-based link addition strategy is applied. In this protection strategy, all the potential links to be added (links in the complement graph) are weighted by the degree of their vertices by $W(d_{ij}) = k_i * k_j$, where k_i and k_j indicate the degree of the vertices in the original network. The strategy is classified as four different strategies and they are introduced as follows.

Non-updated minimum degree link additions

In non-updated minimum degree link addition strategy, the degree weight of each link in the complement network is derived before the process of adding links. Links are added in ascending order according to its degree weight based on the original derived degree weight sequence. In that process, if there are multiple links with the same value of degree weight, they will share the same probability to be added and the algorithm will randomly pick one of them to add.

Non-updated maximum degree link additions

In non-updated maximum degree link addition strategy, the process for adding links is similar to that in non-updated minimum degree link addition strategy. The difference is that, after the the degree weight of each link in the complement network is derived, links are added in descending order according to the original derived degree weight sequence.

Updated minimum degree link additions

In updated minimum degree link addition, the degree weight of each link in the complement network is derived each time before a link is added. The link with the lowest degree weight is then picked to be added. In that process, if there are multiple links with the same value of degree weight, they will share the same probability to be added and the algorithm will randomly pick one of them to add.

Updated maximum degree link additions

The updated maximum degree link addition is similar to the updated maximum degree link addition strategy. The difference is that, each time after the degree weight of each link in the complement network is derived, the link with highest degree weight is picked to be added.

4.1.3. Targeted betweenness-based link additions

The impact of betweenness is also explored. In targeted betweenness-based link addition strategy, links in the complement graph are weighted by the betweenness of their vertices by $W(b_{ij}) = b_i * b_j$, where b_i and b_j indicate the betweenness of the vertices in the original network. The strategy is classified as four types and they are introduced as follows.

Non-updated minimum betweenness link additions

In non-updated minimum betweenness link addition strategy, the betweenness weight of each link in the complement network is derived before the process of adding links.

Links are added in ascending order according to the original derived betweenness weight sequence. In that process, if there are multiple links with the same value of betweenness weight, they will share the same probability to be added and the algorithm will randomly pick one of them to add.

Non-updated maximum betweenness link additions

In non-updated maximum betweenness link addition strategy, the process for adding links is similar to that in non-updated minimum betweenness link addition strategy. The difference is that, after the betweenness weight of each link in the complement network is derived, links are added in descending order according to the original derived betweenness weight sequence.

Updated minimum betweenness link additions

In updated minimum betweenness link addition, the betweenness weight of each link in the complement network is derived each time before a link is added. The link with the lowest betweenness weight is then picked to be added. In that process, if there are multiple links with the same value of betweenness weight, they will share the same probability to be added and the algorithm will randomly pick one of them to add.

Updated maximum betweenness link addition

The updated maximum betweenness link addition is similar to updated maximum betweenness link addition strategy. The difference is that, each time after the betweenness weight of each link in the complement network is derived, the link with the highest betweenness weight is picked to be added.

4.1.4. Targeted eigenvector centrality-based link additions

An exploration is also carried out to understand the significance of eigenvector centrality. In targeted eigenvector centrality-based link addition strategy, links in the complement graph are weighted by the eigenvector centrality of their vertices by $W(x_{ij}) = x_i * x_j$, where x_i and x_j indicate the eigenvector centrality of the vertices in the original network. The strategy is classified as four types and they are introduced as follows.

Non-updated minimum eigenvector centrality link additions

In non-updated minimum eigenvector centrality link addition strategy, the eigenvector centrality weight of each link in the complement network is derived before the process of adding links. Links are added in ascending order according to the original derived eigenvector centrality weight sequence. In that process, if there are multiple links with the same value of eigenvector centrality weight, they will share the same probability to be added and the algorithm will randomly pick one of them to add.

Non-updated maximum eigenvector centrality link additions

In non-updated maximum eigenvector centrality link addition strategy, the process for adding links is similar to that in non-updated minimum eigenvector centrality link addition strategy. The difference is that, after the eigenvector centrality weight of each link in the complement network is derived, links are added in descending order according to the original derived eigenvector centrality weight sequence.

Updated minimum eigenvector centrality link additions

In updated minimum eigenvector centrality link additions, the eigenvector centrality weight of each link in the complement network is derived each time before a link is added. The link with the lowest eigenvector centrality weight is then picked to be added. In that process, if there are multiple links with the same value of eigenvector centrality weight, they will share the same probability to be added and the algorithm will randomly pick one of them to add.

Updated maximum eigenvector centrality link additions

The updated maximum eigenvector centrality link addition is similar to updated maximum eigenvector centrality link addition strategy. The difference is that, each time after the eigenvector centrality weight of each link in the complement network is derived, the link with highest eigenvector centrality weight is picked to be added.

4.2. Node protection strategies

In this section, different node protection strategies are explained. Node protection strategies protect the networks by preventing specific nodes from being attacked. In node protection, only the protection of nodes with high values for certain metrics is considered. This rationale stems from the general principle that nodes with higher metric values tend to hold more prominent and influential roles [19].

4.2.1. Targeted degree-based node protection

The importance of degree is investigated in targeted degree-based node protection strategy. In this protection strategy, the degree of each node in the network is determined before the attack is applied. Based on the derived degree sequences, specific proportions of nodes with highest degree are picked and protected from being attacked. To be specific, each node is protected in descending order according to its degree. In that process, if there are multiple nodes with the same value of degree, they will share the same probability to be protected and the algorithm will randomly pick one of them to be protected.

4.2.2. Targeted betweenness-based node protection

Betweenness is also considered in node protection. In the targeted betweenness-based node protection strategy, the betweenness of each node in the network is derived before the attack is applied. Based on the derived betweenness sequences, specific proportions of nodes with highest betweenness are picked and protected from being attacked. To be specific, each node is protected in descending order according to its betweenness. In that process, if there are multiple nodes with the same value of betweenness, they will share the same probability to be protected and the algorithm will randomly pick one of them to be protected.

4.2.3. Targeted eigenvector centrality-based node protection

Another metric which we do research on is eigenvector centrality. In the targeted eigenvector centrality-based node protection strategy, the eigenvector centrality of each node in the network is derived before the attack is applied. Based on the derived eigenvector centrality sequences, specific proportions of nodes with highest eigenvector centrality are picked and protected from being attacked. To be specific, each node is protected in descending order according to its eigenvector centrality. In that process, if there are multiple nodes with the same value of eigenvector, they will share the same probability to be protected and the algorithm will randomly pick one of them to be protected.

5

Result and analysis

In this chapter, the simulations, analytical approximations and predictions for performance metrics of synthetic networks and real-world networks under different attack strategies, which are introduced in chapter 3 are firstly shown in section 5.1. In section 5.2, two network increasing robustness methods are implemented under the case of updated betweenness attack strategy.

5.1. Robustness assessing results

In this section, all the different attack strategies simulated on Erdős–Rényi graphs, Barabási–Albert graphs, and real-world graphs from the Topology Zoo are demonstrated. Analytical approximations, which can be regarded as a way of prediction, are implemented on the networks under random attacks and stochastic degree attacks. Prediction is also conducted by the machine learning-based approach. It is implemented on networks under random attacks and updated degree attacks.

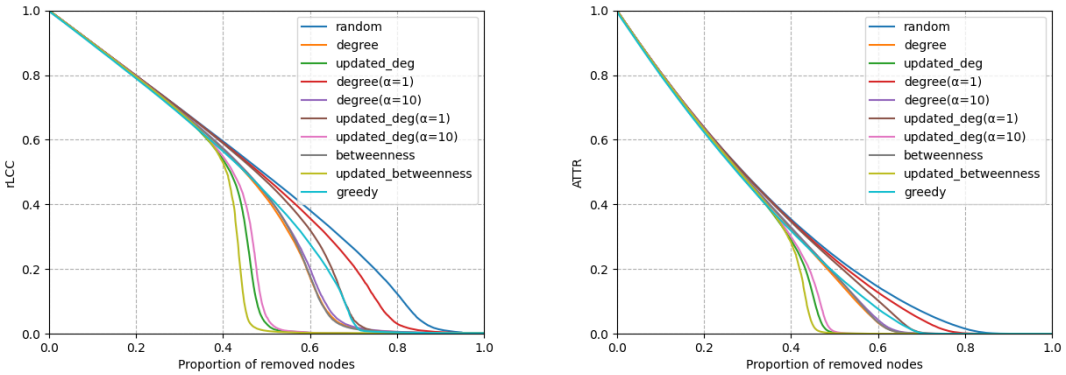
5.1.1. Simulations on synthetic networks

In order to assess the performance of the ten attack strategies, simulations on two different kinds of synthetic graphs are conducted.

ER random graphs

To evaluate the performance of the attack strategies, simulations are conducted on ER random networks with total number of nodes equal to 1000 and the edge formation probability equal to 0.008. One network with specific parameters is firstly generated and the network is then attacked by different strategies. The metric values at each step under attacks is recorded. The final result of each attack strategy is obtained by

averaging the results of 300 generated networks. The average rLCC and ATTR with respect to the proportion of removed nodes for different attack strategies are depicted in Fig. 5.1.



(a) rLCC **(b)** ATTR

Figure 5.1: Comparison of different attack strategies on Erdős–Rényi graphs (1000,0.008). Ten strategies are applied. The x-axis denotes the proportion of nodes removed, the y-axis denotes the performance metrics, which indicate robustness of the network. The blue curves represent the performance of random attacks. The orange curves represent the performance of degree attacks. The green curves represent the performance of updated degree attacks. The red and purple curves represent the performance of stochastic degree attacks with α=1 and with α=10 respectively. The brown and pink curves represent the performance of stochastic updated degree attacks with α=1 and with α=10 respectively. The grey curves represent the performance of betweenness attacks. The olive curves represent the performance of updated betweenness attacks. The cyan curves represent the performance of greedy attacks.

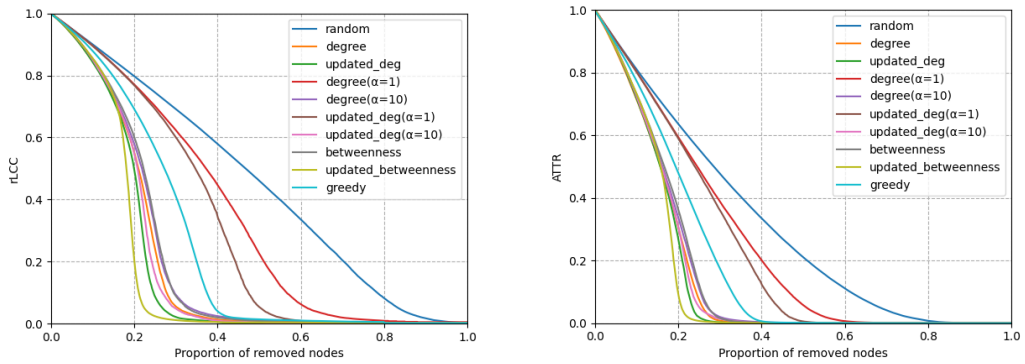
From the figures, it is found that both rLCC and ATTR decrease with increasing number of nodes in the network being removed. To be specific, when one attack strategy is more harmful than another attack strategy in regard to rLCC with a specific proportion of nodes being removed, this attack strategy still does more harm to the network in ATTR with the corresponding proportion of nodes being removed.

Comparing the overall performance of the attack strategies, apart from the stochastic degree attacks, upon Erdős–Rényi graphs, it is found that the performance of updated betweenness attacks are the best, followed by the performance of updated degree attacks. Degree and betweenness attack strategies show similar impacts on the networks, and they perform worse than updated degree attacks. Greedy attack strategy is the second best attack strategy. The difference of the performance between greedy attacks and degree/betweenness attacks becomes larger after about 43% of nodes being removed. Random attacks shows the worst performance. Analysing the stochastic degree attacks, with the increase of parameter α, the strategy becomes more harmful to the network. Besides, the updated attack strategies always outperform the non-updated attack strategies. For all the attack strategies, at the beginning, with that Erdős–Rényi graphs is generated purely random, all the curves coincides with each other. This is because when a node is removed, the remaining nodes still form a fully connected network at the very beginning. In other words, the size of the Largest

Connected Component remains similar as it changes in the previous stage. After approximately 35% of nodes are removed, differences among each strategy become significant.

BA random graphs

Simulations are conducted upon a BA random network with the total number of nodes equal to 300 and the number of edges that a newly added node forms when it joins the network equal to three. One network with specific parameters is firstly generated and the network is then attacked by different strategies. The metric values at each step under attacks is recorded. The final result for each simulation is obtained by averaging the result of 300 networks. The average rLCC and ATTR with respect to the proportion of removed nodes for different attack strategies are depicted in Fig. 5.2.



(a) rLCC **(b)** ATTR

Figure 5.2: Comparison of different attack strategies on Barabási–Albert graphs (500,3). Ten strategies are applied. The x-axis denotes the proportion of nodes removed, the y-axis denotes the performance metrics, which indicate robustness of the network. The blue curves represent the performance of random attacks. The orange curves represent the performance of degree attacks. The green curves represent the performance of updated degree attacks. The red and purple curves represent the performance of stochastic degree attacks with α=1 and with α=10 respectively. The brown and pink curves represent the performance of stochastic updated degree attacks with α=1 and with α=10 respectively. The grey curves represent the performance of betweenness attacks. The olive curves represent the performance of updated betweenness attacks. The cyan curves represent the performance of greedy attacks.

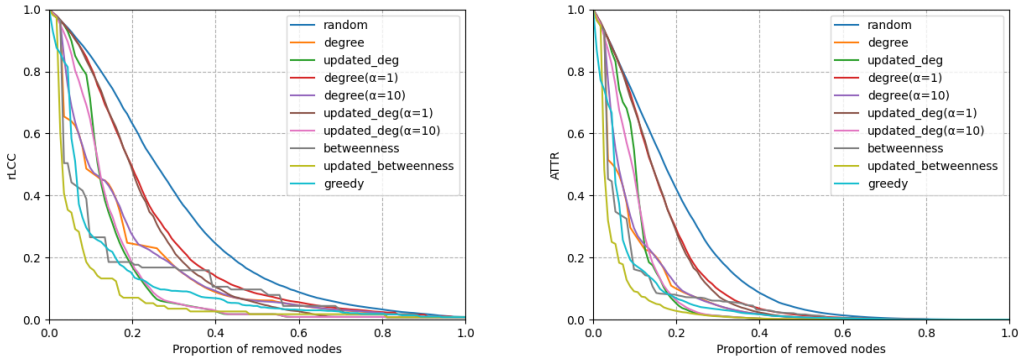
Comparing the performance of different attack strategies on Barabási–Albert graphs, what is different from the observation in Erdős–Rényi graphs case is that, firstly, the curves do not coincide at the beginning. It is due to that Barabási–Albert network model generates networks by iteratively adding nodes and connecting them to existing nodes based on preferential attachment,and it is not a purely random process. In that case, the Largest Connected Component varies with the changes in the attack process. The second difference is that the performance differences between degree attacks and updated degree attacks, betweenness attacks and updated betweenness attacks are smaller in the Barabási–Albert graphs case. To be specific, the updated strategies are much better than the non-updated strategies in Erdős–Rényi graphs

case while the performance of these two types of strategies does not differ much in the Barabási–Albert graphs case. In addition, the degree attack strategy outperforms betweenness attack strategy in the simulation.

5.1.2. Simulations on real-world networks

To assess the performance of the ten attack strategies, simulations on real-world graphs are conducted. The real-world networks used in the simulations are from the Topology Zoo. In this section, firstly, the simulations on a network named 'Deltacom' are analysed. 'Deltacom' is a network with 113 nodes and with degree sequence [0 3 50 31 8 7 4 6 1 1 2]. The sequence gives a concise representation of the distribution of nodes in the network based on their degrees. Each number indicates the number of nodes with a specific degree, starting from degree 0 and incrementing by 1 for each subsequent number in the sequence. Afterwards, to analyse the performance of different attack strategies on real-world networks. The energy metric, which is introduced in Section 2.2, is computed for all networks in the Topology Zoo and then averaged.

When conducting a simulation for one attack strategy, the network is firstly read and then attacked by different strategies. The metric values at each step under attacks is recorded. The final result for each simulation is obtained by averaging the result of 300 networks. The average rLCC and ATTR with respect to the proportion of removed nodes for different attack strategies are depicted in Fig. 5.3.



(a) rLCC **(b)** ATTR

Figure 5.3: Comparison of different attack strategies on network 'Deltacom'. Ten strategies are applied. The x-axis denotes the proportion of nodes removed, the y-axis denotes the performance metrics, which indicate robustness of the network. The blue curves represent the performance of random attacks. The orange curves represent the performance of degree attacks. The green curves represent the performance of updated degree attacks. The red and purple curves represent the performance of stochastic degree attacks with α=1 and with α=10 respectively. The brown and pink curves represent the performance of stochastic updated degree attacks with α=1 and with α=10 respectively. The grey curves represent the performance of betweenness attacks. The olive curves represent the performance of updated betweenness attacks. The cyan curves represent the performance of greedy attacks.

In this case, the updated betweenness attacks still outperform any other attack

strategies. The betweenness and greedy attack strategies perform well before about 20% of nodes are moved, while they are exceeded by degree and updated degree attack afterward. The greedy attack strategy outperforms any other attack strategy initially, but other attack methods surpass its performance after about 3% of nodes are attacked.

To explore the overall performance of the attacks, 233 networks in the Topology Zoo are used and simulations are conducted on them. As it is noticed that when about 80% of nodes are removed, both metrics approach zero, only the metrics before 80% of removed nodes are considered and used to compute the energy. With the energy for each network computed as $E_{i(metric)} = \frac{\sum_{i=1}^{nm} m_i}{N}$, the averaged energy for a specific attack strategy computed as $E_{metric} = \frac{\sum_{i=1}^{n} E_{i(metric)}}{n}$, where n indicates the number of networks. The results are shown in the Table 5.1

	rnd	$ndeg$	$udeg$	$nbet$	$ubet$	$greedy$	$ndeg(\alpha = 1)$	$udeg(\alpha = 1)$	$ndeg(\alpha = 10)$	$udeg(\alpha = 10)$
E_{rLCC}	0.389	0.169	0.149	0.229	0.141	0.166	0.2954	0.2953	0.221	0.201
E_{ATTR}	0.236	0.077	0.070	0.138	0.061	0.075	0.1970	0.1969	0.133	0.12555

Table 5.1: Average energy for networks in the Topology Zoo. In the table, rnd indicates random attacks; $ndeg$ indicates non-updated degree attacks; $udeg$ indicates updated degree attacks; $nbet$ indicates non-updated betweenness attacks; $ubet$ indicates updated-betweenness attacks; $greedy$ indicates greedy attacks; $ndeg(\alpha = 1)$ indicates non-updated stochastic degree attacks for the case when $\alpha = 1$; $ndeg(\alpha = 1)$ indicates updated stochastic degree attacks for the case when $\alpha = 1$; $ndeg(\alpha = 10)$ indicates non-updated stochastic degree attacks for the case when $\alpha = 10$; $udeg(\alpha = 10)$ indicates non-updated stochastic degree attacks for the case when $\alpha = 10$

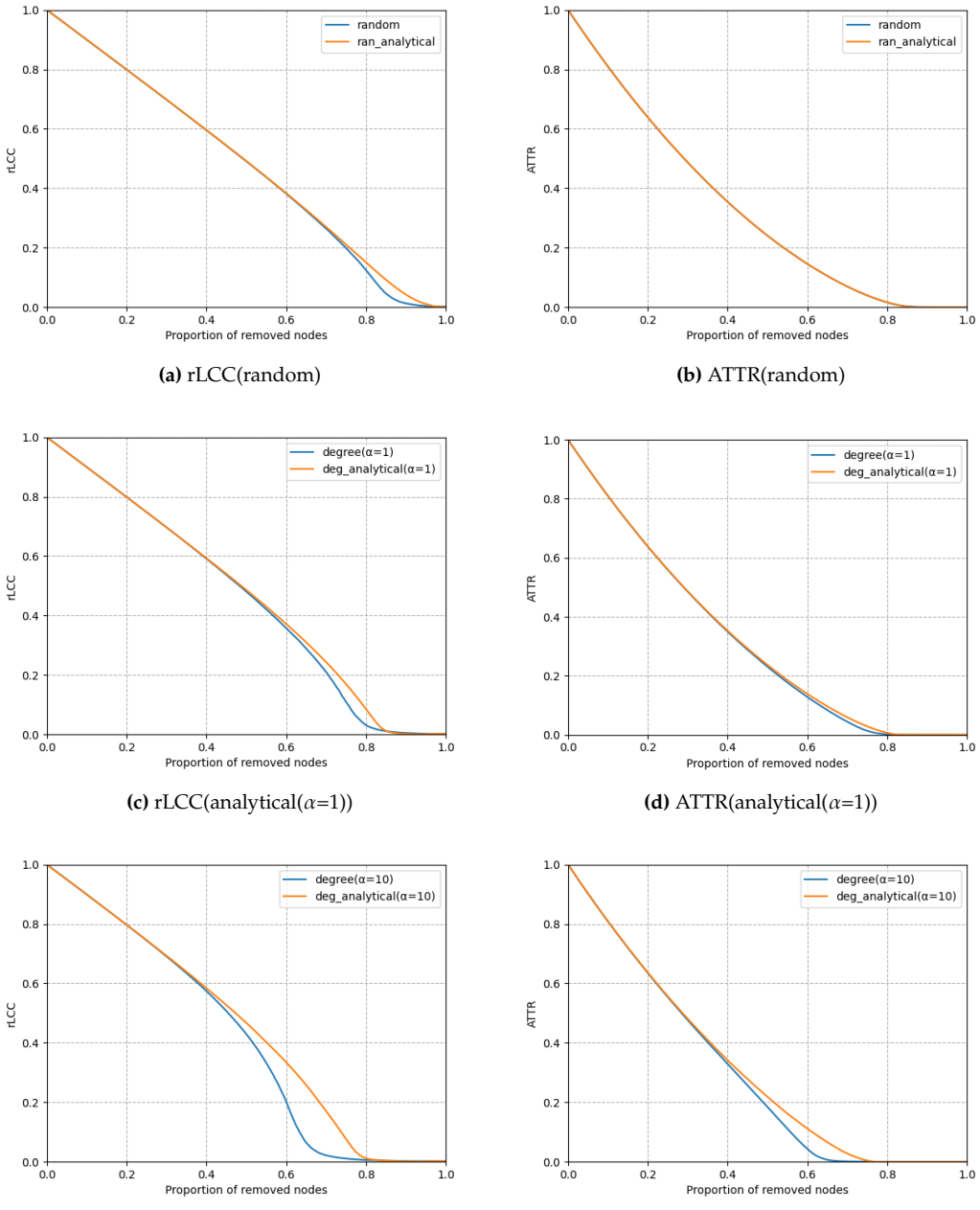
From the table, in both the case of rLCC or ATTR, the averaged energy for random attacks is always the largest, and therefore the random attack process is regarded as the worst strategy. Besides, with the minimum averaged energy occurring in updated betweenness attacks both in the simulations of rLCC and ATTR, updated betweenness attacks are regarded as the best strategy. Generally, higher rLCC energy indicates higher ATTR energy. Based on the results shown in the table, the performance of the attack strategies, from the best to the worst, is ranked as: updated betweenness attacks, updated degree attacks, greedy attacks, non-updated degree attacks, updated stochastic degree attacks with $\alpha = 10$, non-updated stochastic degree attacks with $\alpha = 10$, non-updated betweenness attacks, updated stochastic degree attacks with $\alpha = 1$, non-updated stochastic degree attacks with $\alpha = 1$, random attacks.

5.1.3. Analytical approximations on synthetic networks

In this section, the performance of the analytical approximations is analysed. It includes the approximations under random attacks and the approximations under stochastic degree attacks with α=1 and α=10. The simulations are conducted on Erdős–Rényi graphs, Barabási–Albert graphs, and real-world graphs.

ER random graphs

To evaluate the performance of the analytical approximations, we conduct simulations upon ER random networks with the total number of nodes equal to 1000 and the edge formation probability equal to 0.008. The analytically approximated rLCC and ATTR are depicted in Fig. 5.4 and they are compared with the corresponding simulated results.



(a) rLCC(random)

(b) ATTR(random)

(c) rLCC(analytical(α=1))

(d) ATTR(analytical(α=1))

(e) rLCC(analytical(α=10))

(f) ATTR(analytical(α=10))

Figure 5.4: Evaluation of analytical approximations on Erdős–Rényi graphs (1000,0.008). Three types of attack are applied. The x-axis denotes the proportion of nodes removed, the y-axis denotes the performance metrics, which indicate the robustness of the network. The blue curves represent the simulated results. The orange curves represent the corresponding analytical approximations.

The figures show that the two curves coincide well in the case of random attacks, especially when the algorithm approximates ATTR. The approximation becomes worse when the algorithm approximates the stochastic degree attacks. The difference between the two curves becomes larger when α increases from 1 to 10. Moreover, from the figures, the algorithm can approximate ATTR better than rLCC, as the gap between the curves becomes smaller.

BA random graphs

We conduct simulations upon BA random networks with total number of nodes equal to 500 and the number of edges that a newly added node forms when it joins the network equal to 3. The analytically approximated rLCC and ATTR are depicted in Fig. 5.5 and they are compared with the corresponding simulated results.

The simulations on BA networks show similar results as the simulations on ER networks. The difference is that performance of analytical approximation for stochastic degree attacks becomes worse.

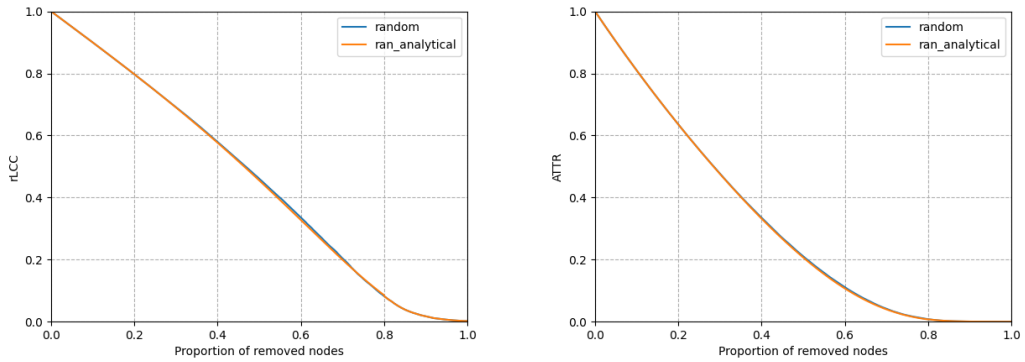
5.1.4. Analytical approximations on real-world networks

To assess the performance of analytical approximations, simulations on real-world graphs are conducted. The real-world networks used in the simulations are from the Topology Zoo. In this section, the simulations on the 'Deltacom' network are analysed. The analytically approximated rLCC and ATTR are depicted in Fig. 5.6 and they are compared with the corresponding simulated results.

An unexpected observation from the results is that the analytical approximations of random attacks perform much worse than what is observed in the case of synthetic networks.

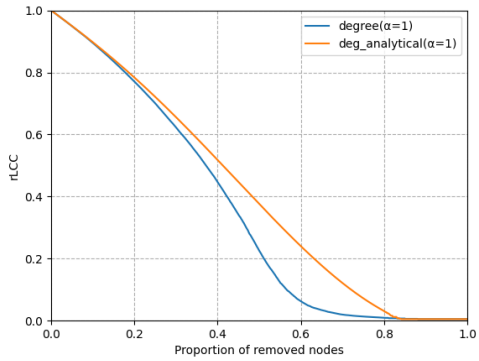
Further simulations to analyse analytical approximations

As it is observed that the analytical approximations do not fit well with the simulated results, the performance of this approach is discussed. It is described in Section 3.2.1 that the analytical approaches are driven by the given generating function, which is derived from the degree sequence. Concisely stated, with a known degree sequence of a network, an approximation of the metrics under different attacks can be computed. It is known that a particular degree sequence can be yielded from a variety of distinct graphs. Therefore, it is considered that the analytical approximations are the averaged results of all the simulations of the networks with the same degree sequence under corresponding attacks. Configuration model graphs are used to generate connected graphs with the same degree sequence. In the simulations, one hundred thousand configuration model graphs with the same degree sequence as the network 'Deltacom' are generated to compare with simulation and analytical approximation results. The comparison is shown in Fig. 5.7.

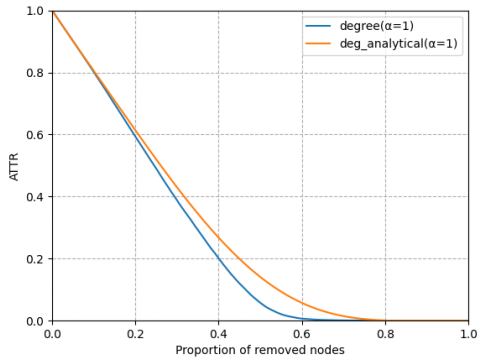


(a) rLCC(random)

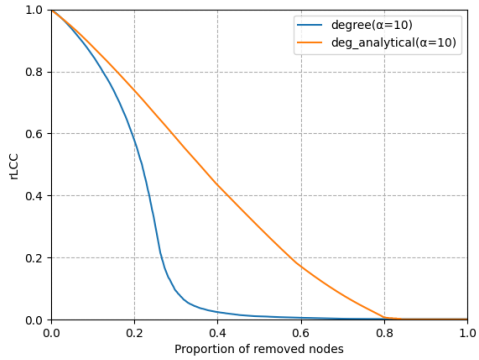
(b) ATTR(random)



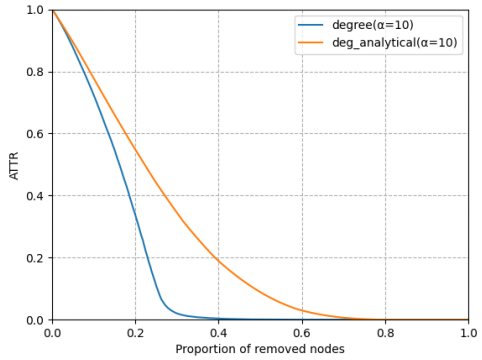
(c) rLCC(analytical(α=1))



(d) ATTR(analytical(α=1))

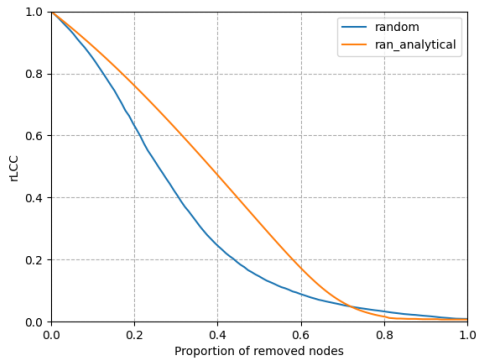


(e) rLCC(analytical(α=10))

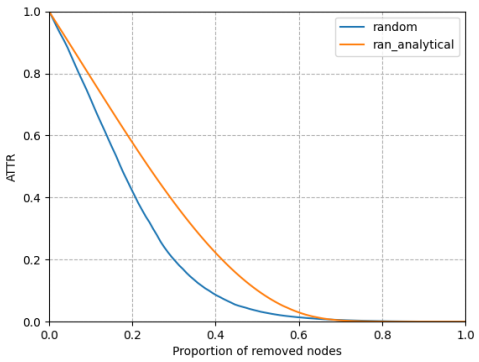


(f) ATTR(analytical(α=10))

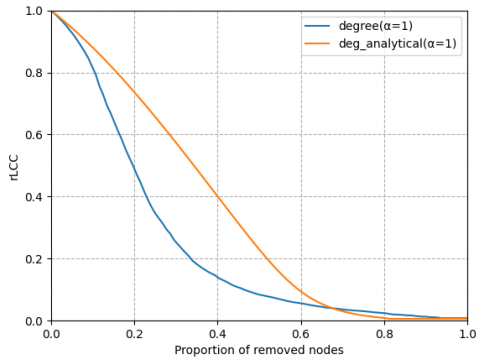
Figure 5.5: Evaluation of analytical approximations on Barabási–Albert graphs (500,3). Three types of attack are applied. The x-axis denotes the proportion of nodes removed, the y-axis denotes the performance metrics, which indicate robustness of the network. The blue curves represent the simulated results. The orange curves represent the corresponding analytical approximations.



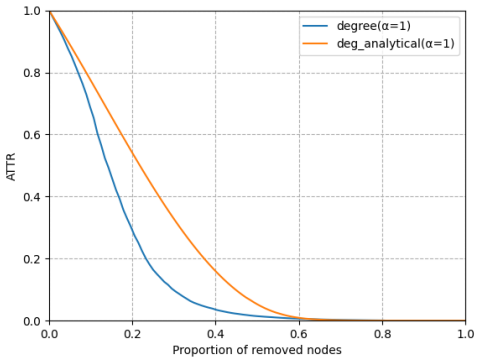
(a) rLCC(random)



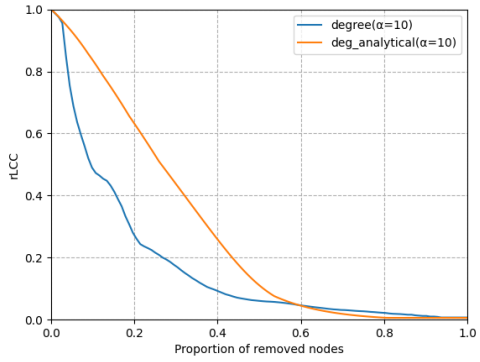
(b) ATTR(random)



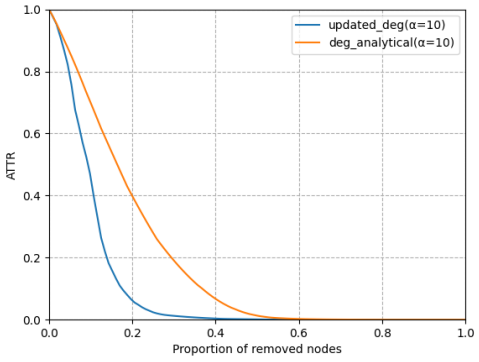
(c) rLCC(analytical(α=1))



(d) ATTR(analytical(α=1))

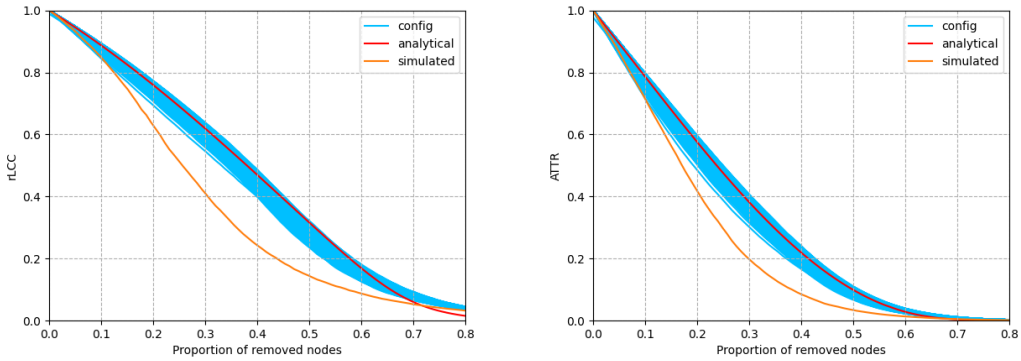


(e) rLCC(analytical(α=10))



(f) ATTR(analytical(α=10))

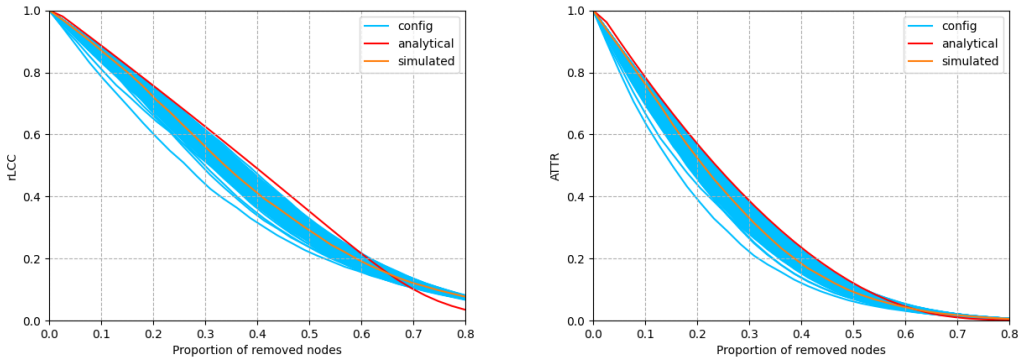
Figure 5.6: Evaluation of analytical approximations on network 'Deltacom'. Three types of attack are applied. The x-axis denotes the proportion of nodes removed, the y-axis denotes the performance metrics, which indicate robustness of the network. The blue curves represent the simulated results. The orange curves represent the corresponding analytical approximations.



(a) rLCC(random) **(b)** ATTR(random)

Figure 5.7: Comparison among simulations on configuration model graphs, simulations and analytical approximations on network 'Deltacom' under random attacks. The x-axis denotes the proportion of nodes removed, the y-axis denotes the performance metrics, which indicate robustness of the network. The blue curves represent the simulation results for the configuration model graphs. The orange curves represent the simulation results. The red curves represent the corresponding analytical approximations.

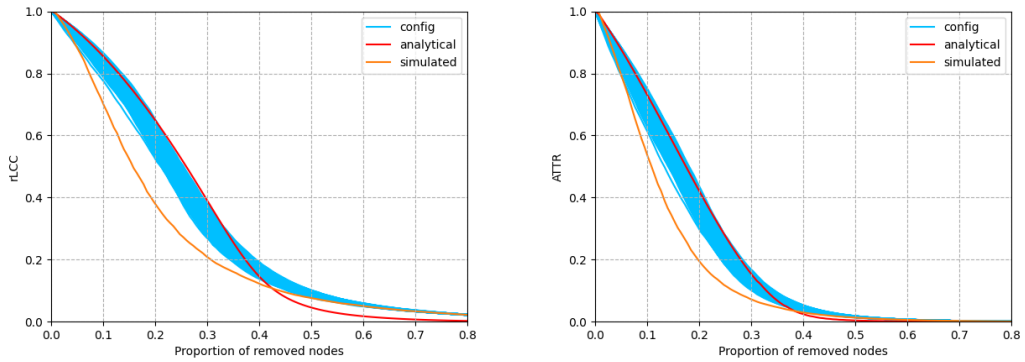
It is observed that the curves for the configuration model graphs include the curves of analytical approximation on both rLCC and ATTR, but do not include the curves of simulated result. To explore the reason, we pick another network 'Geant2012' characterized by a notably distinct degree sequence from the Topology Zoo. It is with degree sequence of [0 8 13 5 5 6 1 1 0 0 1]. Compared with the degree sequence of 'Deltacom' of [0 3 50 31 8 7 4 6 1 1 2], the number of nodes with low degree in 'Geant2012' is much less. The comparison among simulations on configuration model graphs, simulations and analytical approximations on network 'Geant2012' under random attacks is shown in Fig. 5.8.



(a) rLCC(random) **(b)** ATTR(random)

Figure 5.8: Comparison among simulations on configuration model graphs, simulations and analytical approximations on network 'Geant2012' under random attacks. The x-axis denotes the proportion of nodes removed, the y-axis denotes the performance metrics, which indicate robustness of the network. The blue curves represent the simulation results for the configuration model graphs. The orange curves represent the simulation results. The red curves represent the corresponding analytical approximations.

It is observed that the curves for the configuration model graphs include the curves of analytical approximations and the curves of simulated results on both rLCC and ATTR. A preliminary conclusion is made that the configuration model graphs tend to generate networks characterized by a relatively small number of nodes with low degree. To convince this conclusion, another network 'UsCarrier' with the degree sequence [0 8 101 32 14 2 1] from the Topology Zoo is picked. It has 101 nodes with degree of 2, and thus based the previous conclusion drawn, the curves for the configuration model graphs should include the curves of analytical approximation on both rLCC and ATTR, but not include the curves of simulated results, which is of the same as the observation in the results of 'Deltacom'. The comparison among simulations on configuration model graphs, simulations and analytical approximation on network 'UsCarrier' under random attacks is shown in Fig. 5.9. Fig. 5.9 proves the previous



(a) rLCC(random) **(b)** ATTR(random)

Figure 5.9: Comparison among simulations on configuration model graphs, simulations and analytical approximations on network 'UsCarrier' under random attacks. The x-axis denotes the proportion of nodes, the y-axis denotes the performance metrics, which indicate robustness of the network. The blue curves represent the simulation results for the configuration model graphs. The orange curves represent the simulation results. The red curves represent the corresponding analytical approximations.

conclusion made. Besides, an assumption is also made with that it is always observed that the curves for the configuration model graphs include the curves of analytical approximations on both rLCC and ATTR, which is that the analytical approximations are the averaged results of all the simulations of the networks with the same degree sequence under corresponding attacks (random attacks).

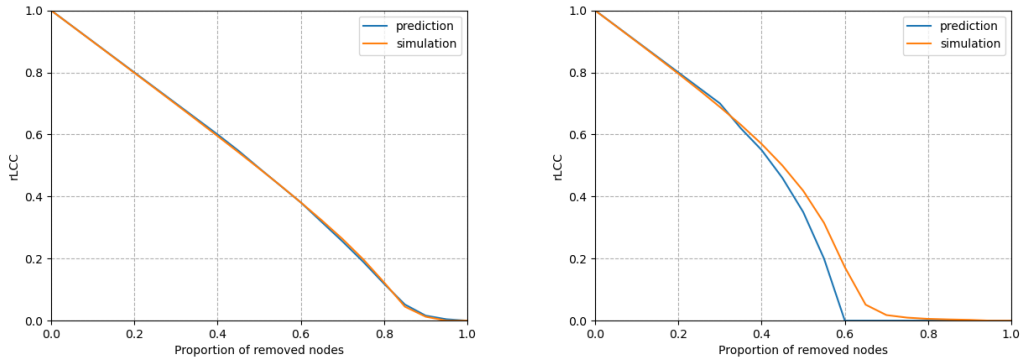
5.1.5. Machine learning based predictions on synthetic networks

This section shows the prediction on synthetic networks using SPP-CNN models under random attacks and updated degree attacks. The simulations are conducted on two Erdős–Rényi networks and a Barabási–Albert network. To train the models, nine representative synthetic network models are chosen to be used as training dataset, including Barabási–Albert (BA) scale-free, extreme homogeneous (EH) , Erdös-Rényi (ER) random-graph , q-snapback (QS) , random hexagon (RH), random triangle (RT), generic scale-free (SF), Newman–Watts small-world (SW-NW), and Watts–Strogatz small-world (SW-WS) network models. The average degree of each

instance of the network is assigned in a reasonably random manner. The average degree range is set as $< k > \in [5,10]$ for the two SW models, $< k > \in [4,8]$ for RH, $< k > \in [3,6]$ for RT, while for the other models $< k > \in [6,12]$. There are two sets of synthetic network models, $S_1 = \{BA, EH, ER, QS, RH, RT, SF, SW-NW, SW-WS\}$, and $S_2 = \{ER, QS, SF, SW-NW\}$. The corresponding network size ranges are set as $N_1 \in [700,1300]$, and $N_2 = \in [300,700]$. For each model, 1000 networks are randomly generated. The adjacency matrix for each network is then used to be part of the training data. The attack simulations are also conducted, and rLCC (as the dataset only contains data of rLCC, only rLCC prediction is considered. ATTR is not considered) in that process is recorded to be the other part of the training data. The iteration time for training the models is set as 20.

ER random graphs

Predictions are conducted upon an ER random network with total number of nodes equal to 1000 and the edge formation probability equal to 0.008. The adjacency matrix of the ER network is obtained and integrated into the model, which has been trained using the training set denoted as S_1. This training set encompasses networks of sizes within the range $N_1 \in [700,1300]$. Subsequently, the model's performance is evaluated through testing. The adjacency matrix of this ER network is derived and is put into the model trained by training set S_1, which is of network size range as $N_1 \in [700,1300]$, to be tested. The predicted rLCC under random attacks and updated degree attacks is depicted in Fig. 5.10 and it is compared with the corresponding simulated result. From the figures, it is found that the two curves coincide well in random attacks. The



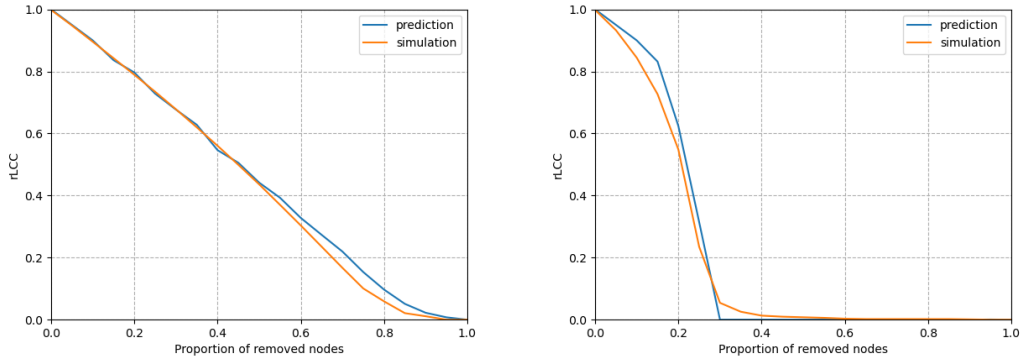
(a) rLCC(random) **(b)** rLCC(updated degree)

Figure 5.10: Predictions on an Erdős–Rényi graph (1000,0.008) under random attacks and updated degree attacks. The x-axis denotes the proportion of nodes removed, the y-axis denotes the performance metric, which indicates robustness of the network. The blue curves represent the predicted results. The orange curves represent the corresponding simulated results.

performance of prediction on updated degree attacks is reasonably adequate.

BA random graphs

Predictions are conducted upon a BA random network with total number of nodes equal to 500 and the number of edges that a newly added node forms when it joins the network equal to 3. The adjacency matrix of the BA network is computed and incorporated into the model that underwent training using the dataset S_2 , where the network sizes fall within the interval $N_1 \in [300,700]$. This model is then subjected to testing to assess its performance. The predicted rLCC under random attacks and updated degree attacks is depicted in Fig. 5.11 and it is compared with the corresponding simulated results. The performance of this model is similar to that of



(a) rLCC(random) **(b)** ATTR(updated degree)

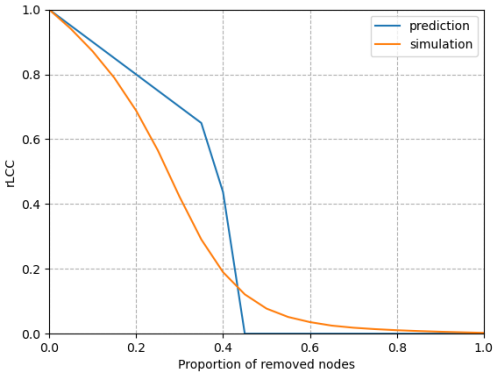
Figure 5.11: Predictions on a Barabási–Albert graph (500,3) under random attacks and updated degree attacks. The x-axis denotes the proportion of nodes removed, the y-axis denotes the performance metric, which indicates robustness of the network. The blue curves represent the predicted results. The orange curves represent the corresponding simulated results.

the model testing ER network. The performance of prediction on random and updated degree attacks is reasonably adequate.

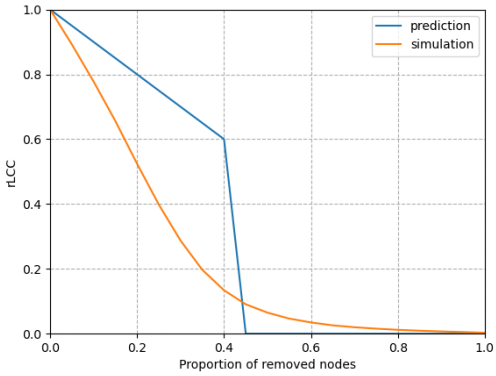
5.1.6. Predictions on real-world networks

This section shows the performance of predictions on real-world networks using SPP-CNN models. The predictions focus on random attacks only. To train the model, 1000 real-world networks are selected randomly and used as training data, denoted as S_r. The network size ranges within $N_r \in [300,700]$. The predictions of rLCC under random attacks are depicted in Fig. 5.12.

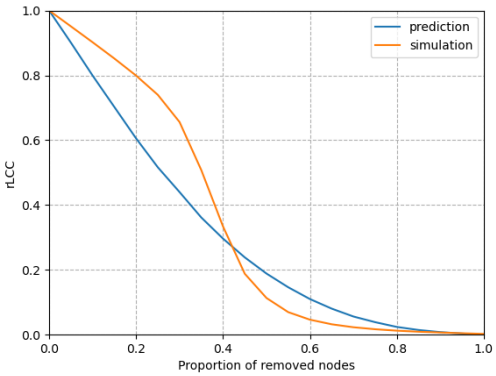
The figures show the results of prediction on real-world networks named 'power-662-bus', 'power-494-bus', 'netz4504-dual' and 'odepa400' from the Network Repository. It is found that there are large gaps between the prediction curves and simulation curves of 'power-662-bus', 'power-494-bus' and 'odepa400', which indicates a bad performance on predictions. We then further explore the prediction performance of the SPP-CNN model on real-world networks by applying the test set. The test set consists of 100 real-world networks with network size ranges within $N_r \in [300,700]$. Based on section 3.2.2, the prediction error ξ is calculated as $\xi = \frac{1}{N}\sum_{i=0}^{N-1}\xi(i)$. With the



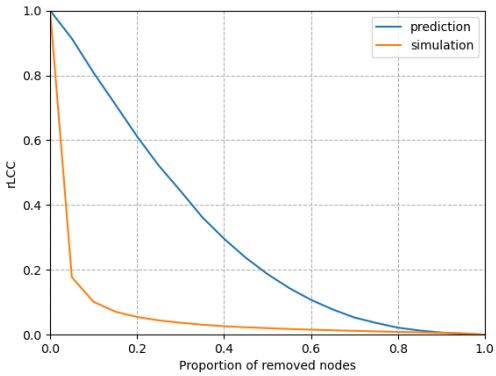
(a) power-662-bus(rLCC, random)



(b) power-494-bus(rLCC, random)



(c) netz4504-dual(rLCC, random)



(d) odepa400(rLCC, random)

Figure 5.12: Predictions on real-world networks under random attacks. The x-axis denotes the proportion of nodes removed, the y-axis denotes the performance metric, which indicates robustness of the network. The blue curves represent the predicted results. The orange curves represent the corresponding simulated results.

test set, the overall prediction errors of the 100 networks are shown in the box plot in Fig. 5.13

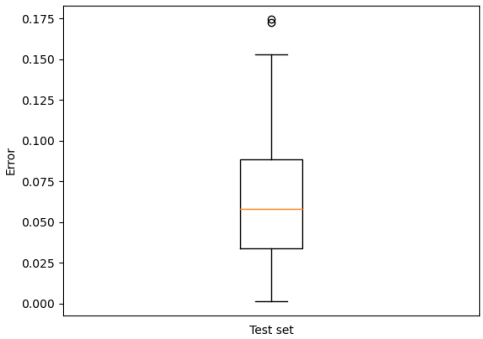
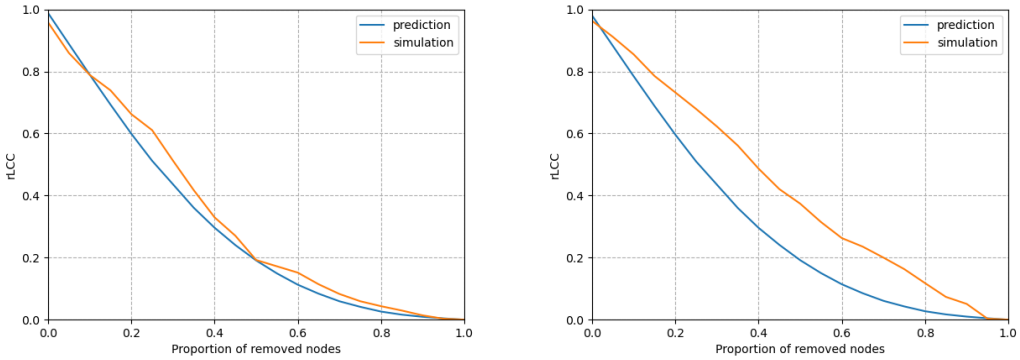


Figure 5.13: Prediction error comparison in the form of a boxplot

The prediction errors indicate the average gaps between the prediction and simulation curves. The box plot shows an acceptable result with that most predicted results are of an error ranging from 0.025 and 0.1. Based on the observation, we define a prediction result with prediction error smaller than 0.025 as a good prediction and a prediction result with prediction error larger than 0.1 as a bad prediction. Fig. 5.14 shows one example in each case with good and bad predictions.



(a) a good prediction example(rLCC)

(b) a bad prediction example(rLCC)

Figure 5.14: Predictions on real-world networks under random attacks. The x-axis denotes the proportion of nodes removed, the y-axis denotes the performance metric, which indicates robustness of the network. The blue curves represent the predicted results. The orange curves represent the corresponding simulated results.

5.2. Robustness increasing results

In this section, all protection strategies are performed on networks under random attack strategy and updated betweenness attack strategy. In this section, we only focus on rLCC, as based on the previous observation, ATTR always shows a similar trend as

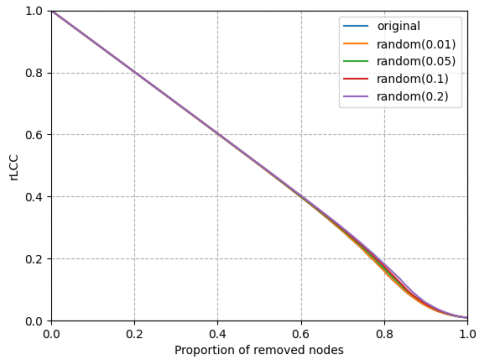
rLCC in the process of nodes being attacked.

5.2.1. Link additions on synthetic networks

In order to assess the performance of the thirteen link addition strategies, we conduct simulations on synthetic networks. To validate the effectiveness of the increasing robustness strategies, we choose random attacks and updated betweenness attacks to test the robustness of networks.

Case 1: under random attacks

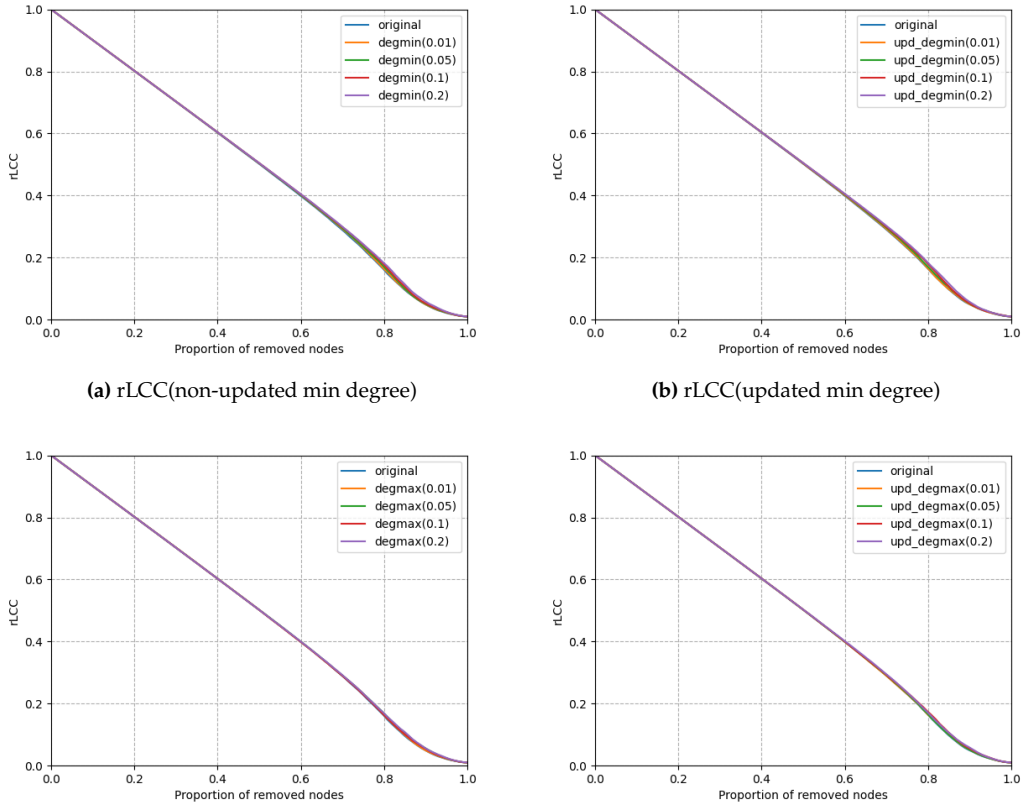
Simulations are conducted on an ER random network with the total number of nodes equal to 100 and the edge formation probability equal to 0.1. In each case, the increasing robustness strategies are conducted before the attacks are simulated. To be specific, links are added according to certain link addition strategies to the original ER random networks at first, and then the newly constructed network is attacked. Four cases are explored for each link addition strategy, where 1%, 5%, 10% and 20% of links are added. The final result for each simulation is obtained by averaging the results of 300 networks after being protected and then attacked. The average rLCC with respect to the proportion of removed nodes for different link addition strategies are depicted in Fig. 5.15, Fig. 5.16, Fig. 5.17, and Fig. 5.18.



(a) rLCC(random)

Figure 5.15: Performance of random link additions on Erdős–Rényi graphs (100,0.1) under random attacks. The x-axis denotes the proportion of nodes removed, the y-axis denotes rLCC. The blue curves represent the case when no link is added. The blue curves represent the case when no link is added. The orange, green, red and purple curves represent the case when 1%, 5%, 10% and 20% of links are added respectively.

From the figures, it is observed that all the link addition strategies do not exert a significant impact on the robustness of the networks under random attacks with the observation that all the five curves almost coincide with each other through the whole process in each case. After about 70% of nodes are removed, the impact of link addition strategy becomes slightly more significant with more links being added. Based on the observation, it is considered that the ER network already exhibits a degree



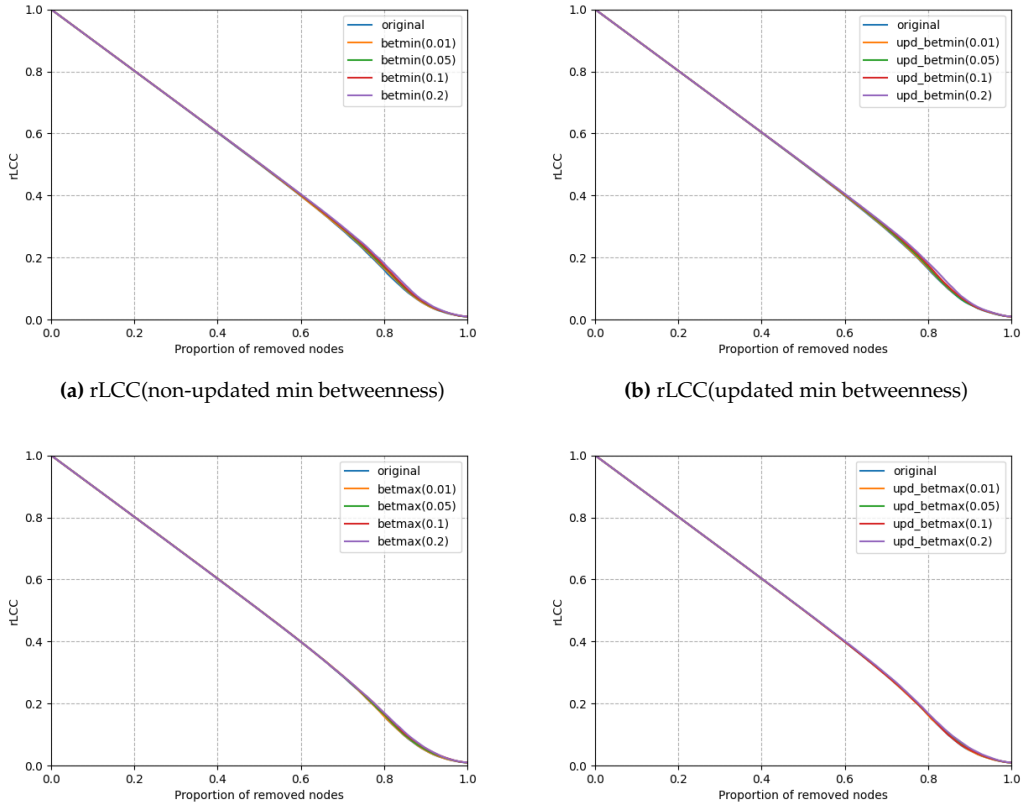
(a) rLCC(non-updated min degree)

(b) rLCC(updated min degree)

(c) rLCC(non-updated max degree)

(d) rLCC(updated max degree)

Figure 5.16: Comparison of targeted degree-based link additions on Erdős–Rényi graphs (100,0.1) under random attacks. The x-axis denotes the proportion of nodes removed, the y-axis denotes rLCC. The blue curves represent the case when no link is added. The blue curves represent the case when no link is added. The orange, green, red and purple curves represent the case when 1%, 5%, 10% and 20% of links are added respectively.



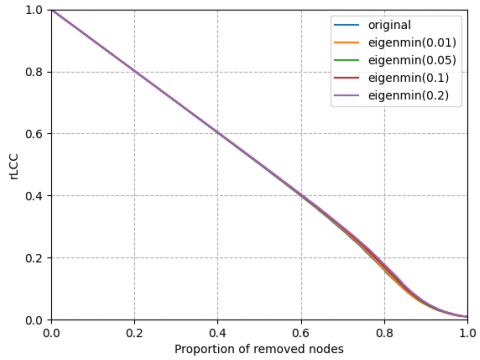
(a) rLCC(non-updated min betweenness)

(b) rLCC(updated min betweenness)

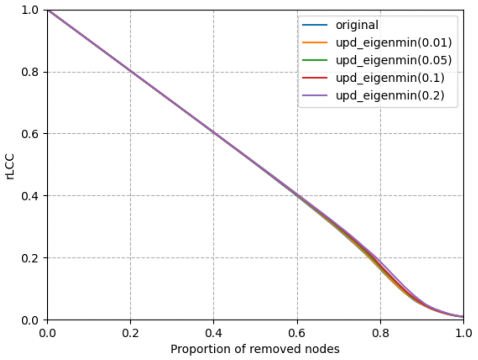
(c) rLCC(non-updated max betweenness)

(d) rLCC(updated max betweenness)

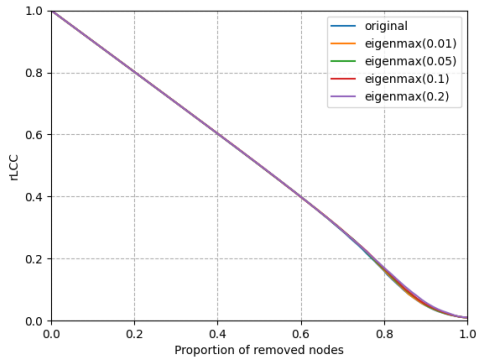
Figure 5.17: Comparison of targeted betweenness-based link additions on Erdős–Rényi graphs (100,0.1) under random attacks. Thirteen strategies are applied. The x-axis denotes the proportion of nodes removed, the y-axis denotes rLCC. The blue curves represent the case when no link is added. The blue curves represent the case when no link is added. The orange, green, red and purple curves represent the case when 1%, 5%, 10% and 20% of links are added respectively.



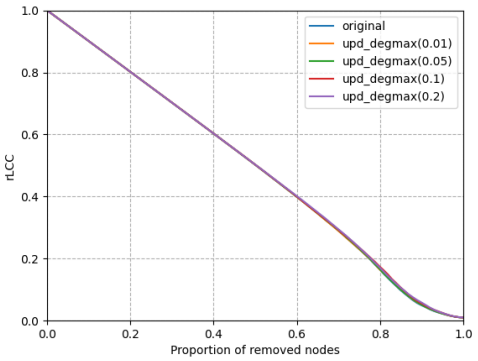
(a) rLCC(non-updated min eigenvector centrality)



(b) rLCC(updated min eigenvector centrality)



(c) rLCC(non-updated max eigenvector centrality)



(d) rLCC(updated max eigenvector centrality)

Figure 5.18: Comparison of targeted eigenvector centrality-based link additions on Erdős–Rényi graphs (100,0.1) under random attacks. Thirteen strategies are applied. The x-axis denotes the proportion of nodes removed, the y-axis denotes rLCC. The blue curves represent the case when no link is added. The blue curves represent the case when no link is added. The orange, green, red and purple curves represent the case when 1%, 5%, 10% and 20% of links are added respectively.

of robustness under random attacks due to the random nature of edge formation, making it less sensitive to additional links. To be specific, in a random attack scenario where nodes are removed randomly, the attack strategy does not specifically target nodes with added links. The random nature of the attack means that any node in the network, whether it has additional links or not, has an equal chance of being removed. Therefore, the added links may not provide targeted protection to the most critical nodes.

Though the performance of link addition strategies in the scenario of random attacks is unsatisfactory, it is noticed that the performance of different protection strategies still varies. Therefore we explore the overall performance of each link addition strategy by calculating the energy in each case. The computed energy in terms of different proportions of links added is shown in the Table 5.2.

	org	rnd	$ndmin$	$udmin$	$ndmax$	$udmax$	$nbmin$	$ubmin$	$nbmax$	$ubmax$	$nemin$	$uemin$	$nemax$	$uemax$
$E_{rLCC}(0.01)$	0.5433	0.5436	0.5443	0.5441	0.5433	0.5431	0.5443	0.5447	0.5437	0.5435	0.5444	0.5443	0.5440	0.5433
$E_{rLCC}(0.05)$	0.5433	0.5448	0.5452	0.5458	0.5436	0.5439	0.5455	0.5457	0.5443	0.5441	0.5455	0.5458	0.5436	0.5434
$E_{rLCC}(0.1)$	0.5433	0.5457	0.5470	0.5472	0.5442	0.5447	0.5463	0.5472	0.5436	0.5449	0.5463	0.5472	0.5436	0.5441
$E_{rLCC}(0.2)$	0.5433	0.5480	0.5480	0.5491	0.5450	0.5461	0.5481	0.5489	0.5454	0.5452	0.5475	0.5489	0.5451	0.5449

Table 5.2: Energy for ER networks. In the table, org indicates the case with no link addition strategy applied; rnd indicates random link additions; $ndmin$ indicates non-updated minimum degree link additions; $udmin$ indicates updated minimum degree link additions; $ndmax$ indicates non-updated maximum degree link additions; $udmax$indicates updated maximum degree link additions; $nbmin$ indicates non-updated minimum betweenness link additions; $ubmin$ indicates updated minimum betweenness link additions; $nbmax$ indicates non-updated maximum betweenness link additions; $ubmax$ indicates updated maximum betweenness link addition; $nemin$ indicates non-updated minimum eigenvector centrality link additions; $uemin$ indicates updated minimum eigenvector centrality link additions; $nemax$ indicates non-updated maximum eigenvector centrality link additions; $uemax$ indicates updated maximum eigenvector centrality link additions.

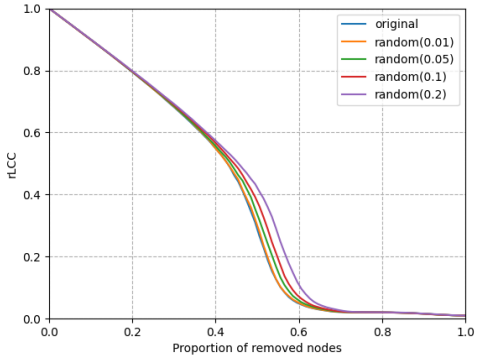
From the table, for the case where 1% of total links are added, the updated minimum betweenness link addition strategy shows the best protecting performance with the highest energy observed. The worst performance occurs in updated maximum degree link addition strategy, with value of energy equal to 0.5431. For the case where 5% of total links are added, updated minimum degree and updated minimum eigenvector centrality link addition strategies show the best protecting performance, while the lowest energy is found in updated maximum eigenvector centrality link addition strategy. For the case where 10% of total links are added, the three updated minimum certain metric link addition strategies all show the best performance, and the non-updated maximum eigenvector centrality link addition strategy shows the worst performance. For the case where 20% of total links are added, updated minimum degree link addition strategy shows the best protecting performance,and the updated maximum eigenvector centrality link addition strategy shows the worst performance.

To summarize, minimum link addition strategies always outperform the maximum strategies, with the fact that the best protecting strategies are always found in minimum link addition strategies, and the worst protecting strategies are always found in maximum link addition strategies. Because the values of the energy differ slightly

in each case, to convince the summary and to find new observations, we further investigate the performance of these link addition strategies for the networks under updated betweenness attacks. The reason why this attack strategy is picked is that the updated betweenness attacks are the most harmful attack strategy based on the observations in previous sections.

Case 2: under updated betweenness attacks

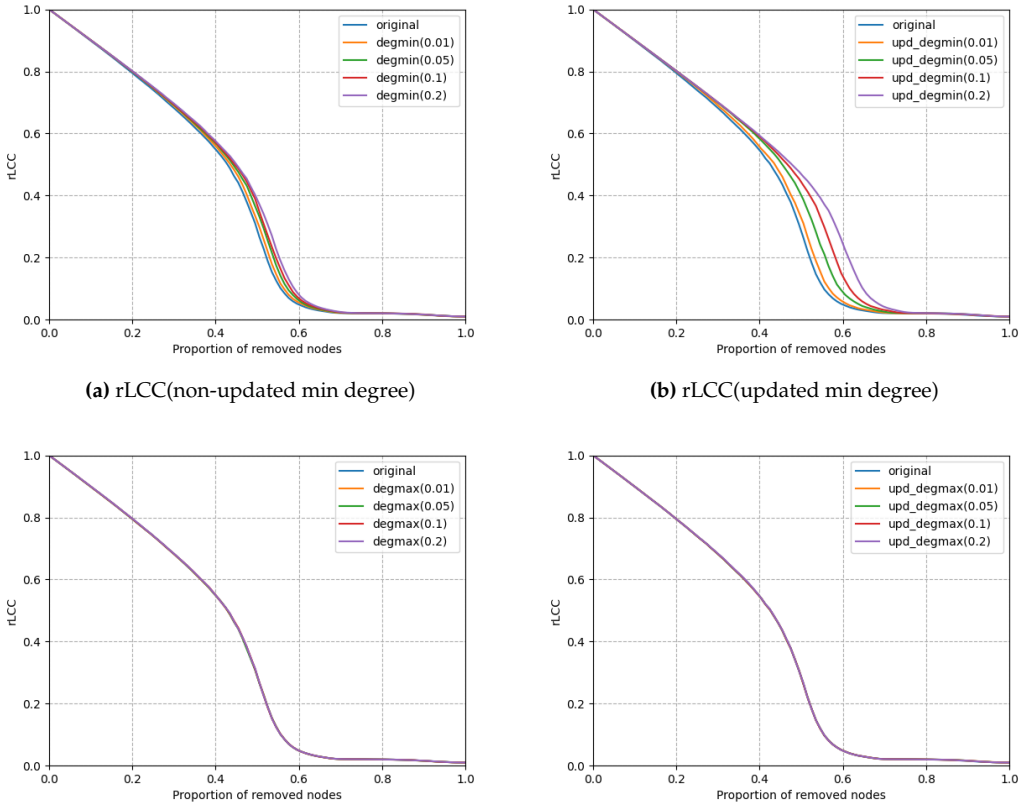
To further investigate the performance of the thirteen protecting strategies, simulations are conducted on the ER random networks with total number of nodes equal to 100 and the edge formation probability equal to 0.1. The simulation processes are the same as what is done in case 1, except for that the networks are under updated betweenness attacks after their robustness is increased through certain approaches. The final result for each simulation is obtained by averaging the result of 300 networks after being protected and then attacked. The average rLCCs with respect to the proportion of removed nodes for different link addition strategies are depicted in Fig. 5.19, Fig. 5.20, Fig. 5.21, and Fig. 5.22.



(a) rLCC(random)

Figure 5.19: Performance of random link additions on Erdős–Rényi graphs (100,0.1) under updated betweenness attacks. The x-axis denotes the proportion of nodes removed, the y-axis denotes rLCC. The blue curves represent the case when no link is added. The blue curves represent the case when no link is added. The orange, green, red and purple curves represent the case when 1%, 5%, 10% and 20% of links are added respectively.

Compared with the figures in case 1, in which networks are under random attacks, it is found that the influence of the link addition strategies becomes more significant with the observation of the gaps among the five curves in some cases. It is clear from the figures that the maximum certain metric link addition strategies perform much worse than the minimum certain metric link addition strategies, as it is noticed that the five curves almost coincide with each other with the maximum certain metric link addition strategies applied. It indicates that the protection barely increases the robustness of the networks. This is because the minimum certain metric link addition strategies help to increase the degree, which also generally indicates an increase in betweenness, for the nodes with low certain metrics. Meanwhile, the strategies do no protection on the nodes which are already crucial enough and therefore still have



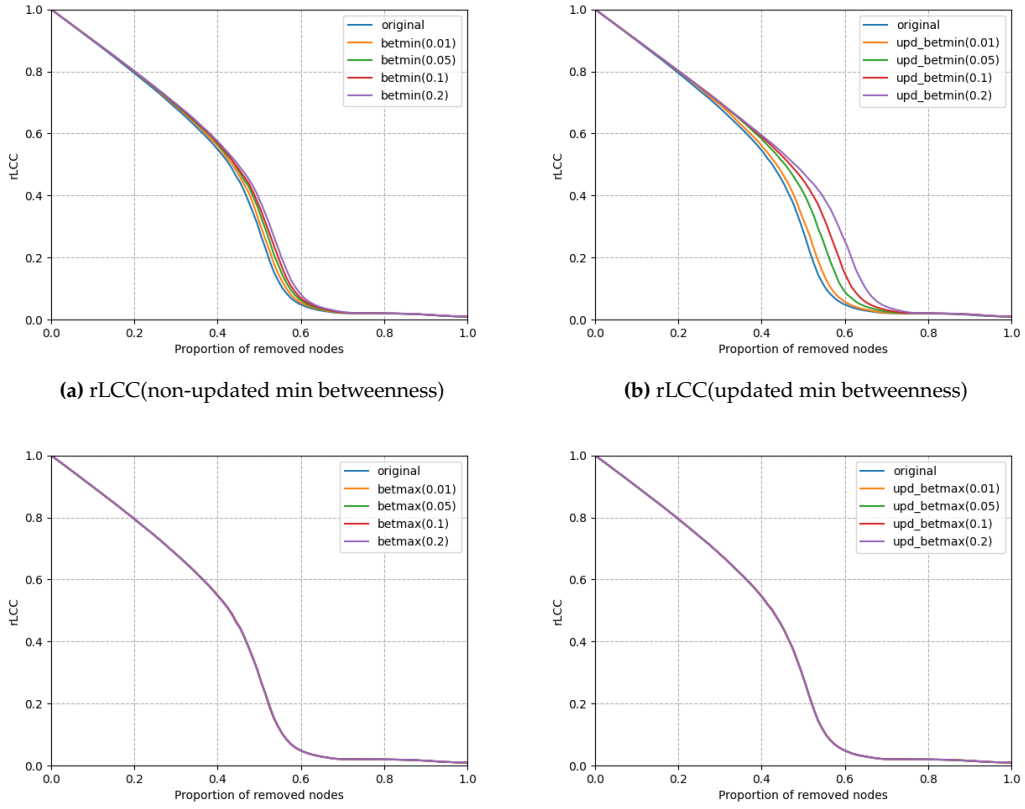
(a) rLCC(non-updated min degree)

(b) rLCC(updated min degree)

(c) rLCC(non-updated max degree)

(d) rLCC(updated max degree)

Figure 5.20: Comparison of targeted degree-based link additions on Erdős–Rényi graphs (100,0.1) under updated betweenness attacks. The x-axis denotes the proportion of nodes removed, the y-axis denotes rLCC. The blue curves represent the case when no link is added. The blue curves represent the case when no link is added. The orange, green, red and purple curves represent the case when 1%, 5%, 10% and 20% of links are added respectively.



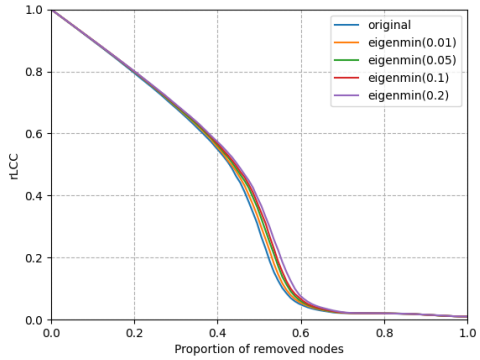
(a) rLCC(non-updated min betweenness)

(b) rLCC(updated min betweenness)

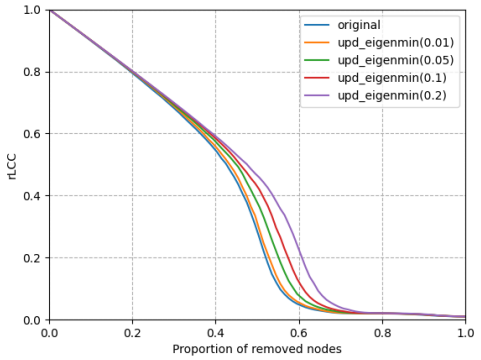
(c) rLCC(non-updated max betweenness)

(d) rLCC(updated max betweenness)

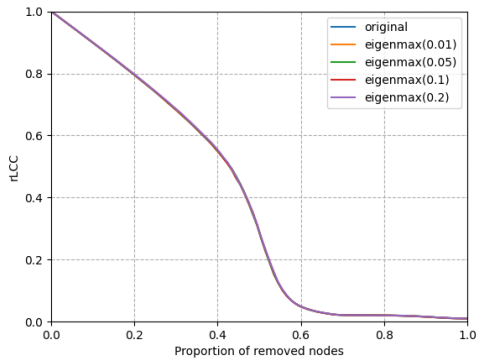
Figure 5.21: Comparison of targeted betweenness-based link additions on Erdős–Rényi graphs (100,0.1) under updated betweenness attacks. The x-axis denotes the proportion of nodes removed, the y-axis denotes rLCC. The blue curves represent the case when no link is added. The blue curves represent the case when no link is added. The orange, green, red and purple curves represent the case when 1%, 5%, 10% and 20% of links are added respectively.



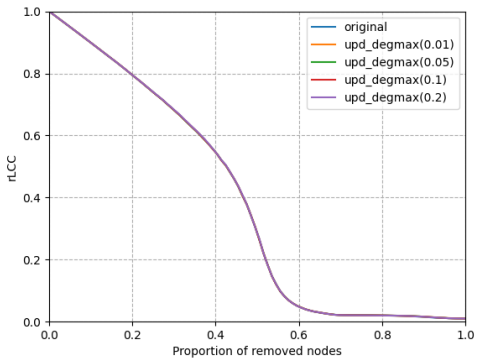
(a) rLCC(non-updated min eigenvector centrality)



(b) rLCC(updated min eigenvector centrality)



(c) rLCC(non-updated max eigenvector centrality)



(d) rLCC(updated max eigenvector centrality)

Figure 5.22: Comparison of targeted eigenvector centrality-based link additions on Erdős–Rényi graphs (100,0.1) under updated betweenness attacks. The x-axis denotes the proportion of nodes removed, the y-axis denotes rLCC. The blue curves represent the case when no link is added. The orange, green, red and purple curves represent the case when 1%, 5%, 10% and 20% of links are added respectively.

high possibility to be removed under updated betweenness attacks. Though the originally more important nodes are removed under the updated betweenness attack, the remaining nodes become more important because of the link addition strategies. In that case, the networks become more robust with the centrality of the nodes, which are originally less important and are less likely to be removed, increases. On the contrary, the maximum certain metric link addition strategies always increase the centrality of nodes which are already important. After protection is conducted, the nodes with high centrality are still of higher possibility to be attacked, and in that case, the strategies do not sufficiently protect the networks.

Comparing figures in the scenarios with updated and non-updated minimum certain metric link addition strategies, with larger gaps found between the curves indicating networks without protections and the curves indicating networks under protection, it is concluded that the updated strategies always outperform the non-updated strategies. The reason is that the updated strategies exhibit a superior capacity to enhance the centrality of less significant nodes when compared to the non-updated strategies.

To compare the overall performance, energy is calculated in each case. The computed energy in terms of different proportions of links added is shown in the Table 5.3.

	org	rnd	$ndmin$	$udmin$	$ndmax$	$udmax$	$nbmin$	$ubmin$	$nbmax$	$ubmax$	$nemin$	$uemin$	$nemax$	$uemax$
$E_{rLCC}(0.01)$	0.4232	0.4249	0.4299	0.4321	0.4232	0.4232	0.4304	0.4323	0.4233	0.4232	0.4286	0.4300	0.4231	0.4236
$E_{rLCC}(0.05)$	0.4232	0.4310	0.4353	0.4491	0.4233	0.4234	0.4352	0.4510	0.4233	0.4232	0.4333	0.4435	0.4235	0.4240
$E_{rLCC}(0.1)$	0.4232	0.4380	0.4392	0.4614	0.4233	0.4233	0.4391	0.4629	0.4232	0.4233	0.4372	0.4564	0.4236	0.4242
$E_{rLCC}(0.2)$	0.4232	0.4508	0.4450	0.4763	0.4235	0.4234	0.4448	0.4774	0.4233	0.4232	0.4426	0.4734	0.4248	0.4253

Table 5.3: Energy for ER networks. In the table, org indicates the case with no link addition strategy applied; rnd indicates random link additions; $ndmin$ indicates non-updated minimum degree link additions; $udmin$ indicates updated minimum degree link additions; $ndmax$ indicates non-updated maximum degree link additions; $udmax$indicates updated maximum degree link additions; $nbmin$ indicates non-updated minimum betweenness link additions; $ubmin$ indicates updated minimum betweenness link additions; $nbmax$ indicates non-updated maximum betweenness link additions; $ubmax$ indicates updated maximum betweenness link addition; $nemin$ indicates non-updated minimum eigenvector centrality link additions; $uemin$ indicates updated minimum eigenvector centrality link additions; $nemax$ indicates non-updated maximum eigenvector centrality link additions; $uemax$ indicates updated maximum eigenvector centrality link additions.

As it is analysed that all of the link addition strategies based on maximum certain metrics do not effectively protect the networks, and that they all show close performance, it is concluded that the worst attack strategies are not necessary to be discussed. From the table, for all the cases, it is found that the updated minimum betweenness link addition strategy always demonstrates superior network protection performance with the highest value of energy. Therefore, it is concluded that the updated minimum betweenness link addition strategy is the best protection strategy when the networks are under updated betweenness attacks.

5.2.2. Link additions on real-world networks

After the performance of different link addition strategies on synthetic networks is evaluated and analysed, we would also like to check the performance of the strategies on real-world networks. It is acknowledged that the performance of the strategies would vary with that the topology of real-world networks differs. In this section, we only focus on link addition strategies with 5% and 20% of links added, and a network named 'Garr201103' to investigate if maximum certain metric link addition strategies still perform much worse than the minimum certain metric link addition strategies. In the simulations, we also would like to evaluate the performance of random link addition strategy. It is assumed that the network is under random attacks and updated betweenness attacks.

Case 1: under random attacks

Simulations are conducted on the network named 'Garr201103' from the Topology Zoo. 'Garr201103' is a network with 58 nodes and with degree sequence [0 33 9 2 5 1 3 2 1 0 1 0 1]. The final result for each simulation is obtained by averaging the result of 300 'Garr201103' networks after being protected and then attacked. All the link addition strategies are simulated. The average rLCC with respect to the proportion of removed nodes for the case of 5% links being added is depicted in Fig. 5.23.

In the cases where 5% links are added, the orange, green and red curves coincide with each other in each plot, and thus it is hard to distinguish the performance. However, even with only 5% links being added, there are noticeable increases on the robustness of the networks.

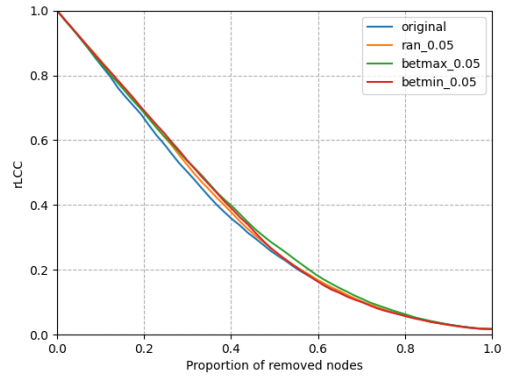
The average rLCC with respect to the proportion of removed nodes for the case of 20% links being added is depicted in Fig. 5.24.

In the case where 20% links are added, it is noticed that, for most of the time in the process of node removals, the red curves are above the green curves, which indicates that the performance of the minimum certain metric link addition strategies outperforms that of the maximum certain metric link addition strategies. It is interesting that the random link addition strategy is the a good strategy with the rLCC in which is the highest in most cases. The updated minimum betweenness link addition strategy is still the best protection strategy when the network is under random attacks.

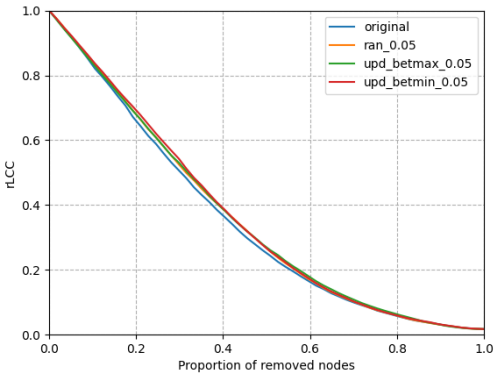
Case 2: under updated betweenness attacks

The network named 'Garr201103' now faces updated betweenness attacks. With all the link addition strategies being simulated, we firstly focus on the case of 5% links being added. The results are depicted in Fig. 5.25.

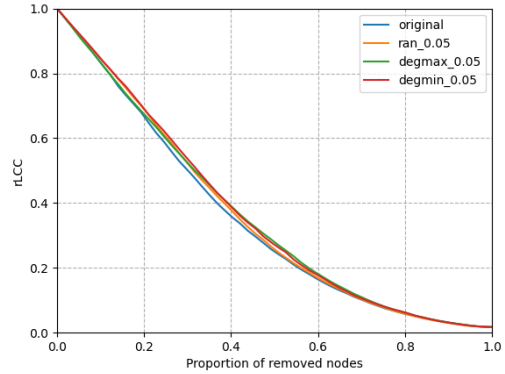
It is observed that there are noticeable increases on the robustness of the networks with only 5% links being added.



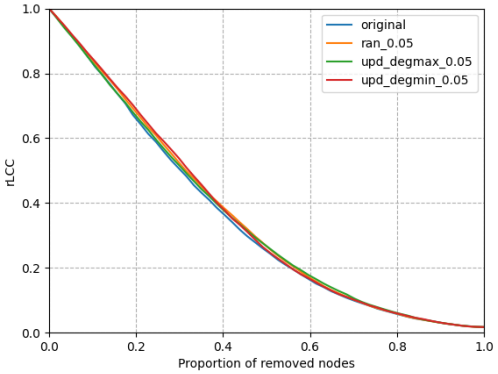
(a) rLCC(non-updated betweenness(0.05))



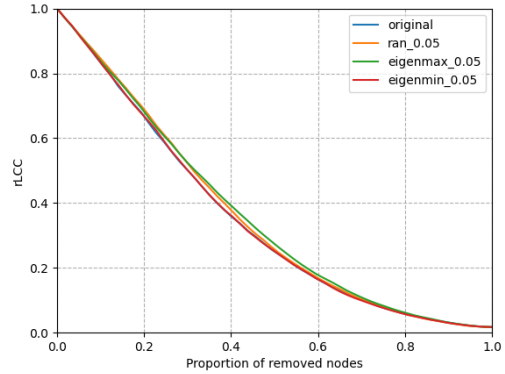
(b) rLCC(updated betweenness(0.05))



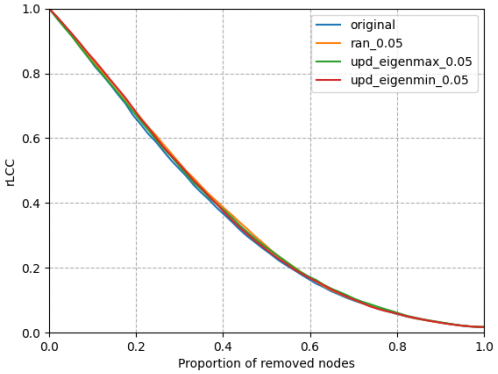
(c) rLCC(non-updated degree(0.05))



(d) rLCC(updated degree(0.05))

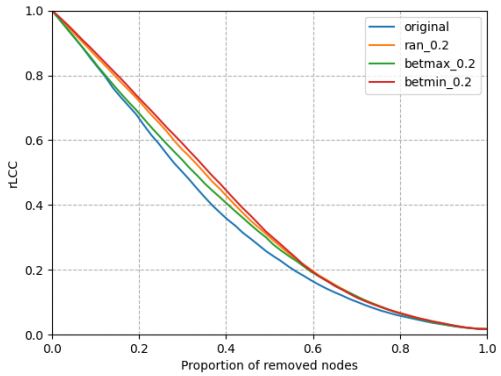


(e) rLCC(non-updated eigenvector centrality(0.05))

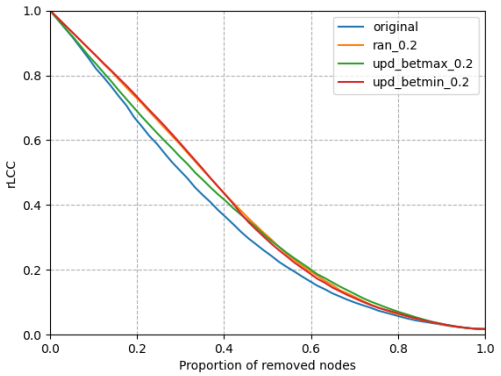


(f) rLCC(updated eigenvector centrality(0.05))

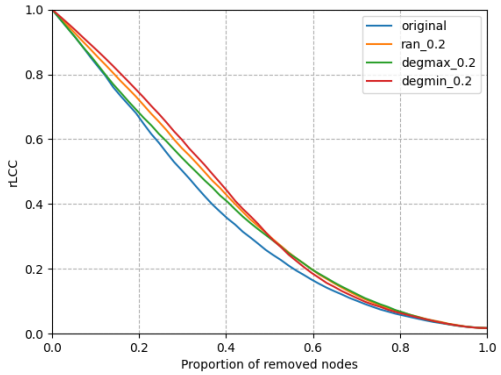
Figure 5.23: Comparison of different link addition strategies on a real-world network named 'Garr201103' under random attacks with 5% of links added. The x-axis denotes the proportion of nodes removed, the y-axis denotes the performance metric, rLCC to be specific, which indicates robustness of the network. The blue curves represent the case when no link is added. The orange curves represent the results of random link addition strategies. The green and red curves represent the results of maximum and minimum link addition strategies respectively.



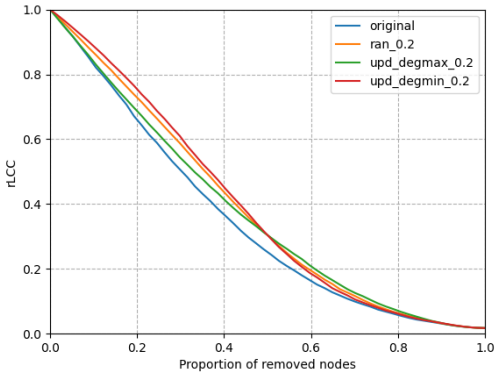
(a) rLCC(non-updated betweenness(0.2))



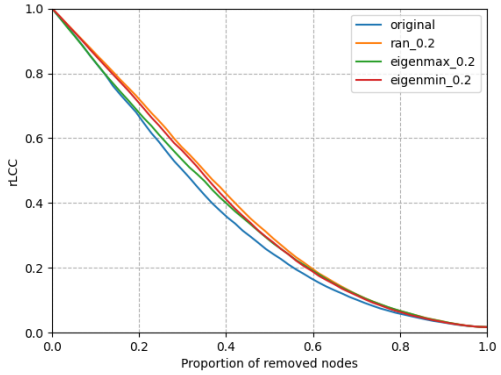
(b) rLCC(updated betweenness(0.2))



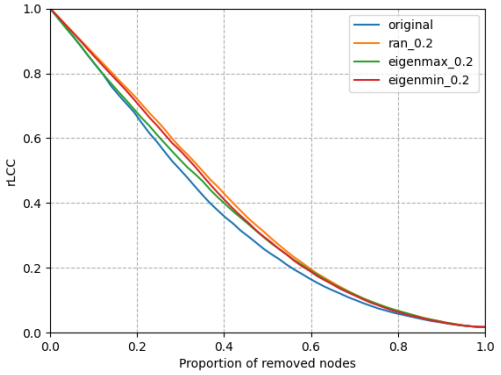
(c) rLCC(non-updated degree(0.2))



(d) rLCC(updated degree(0.2))

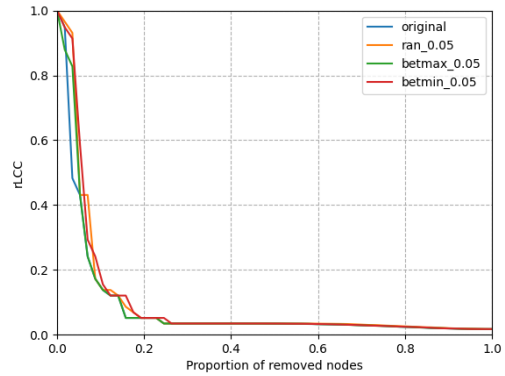


(e) rLCC(non-updated eigenvector centrality(0.2))

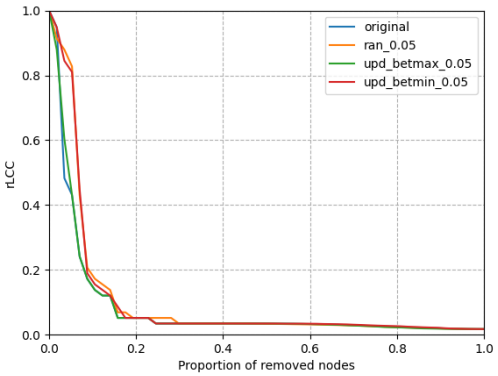


(f) rLCC(updated eigenvector centrality(0.2))

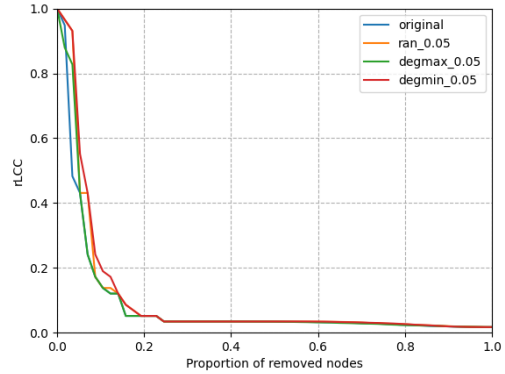
Figure 5.24: Comparison of different link addition strategies on a real-world network named 'Garr201103' under random attack with 20% of links added. The x-axis denotes the proportion of nodes removed, the y-axis denotes the performance metric, rLCC to be specific, which indicates robustness of the network. The blue curves represent the case when no link is added. The orange curves represent the results of random link addition strategies. The green and red curves represent the results of maximum and minimum link addition strategies respectively.



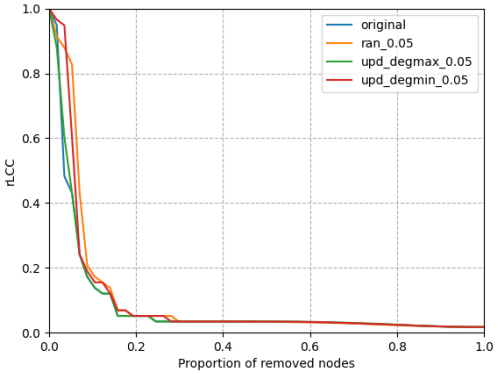
(a) rLCC(non-updated betweenness(0.05))



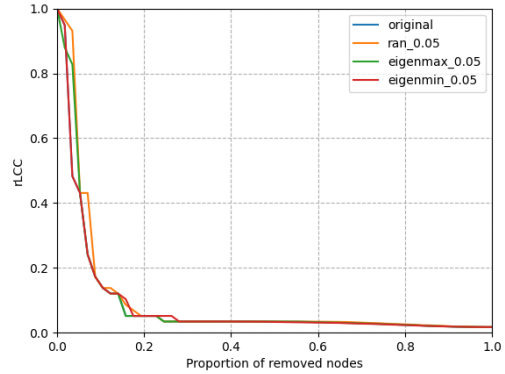
(b) rLCC(updated betweenness(0.05))



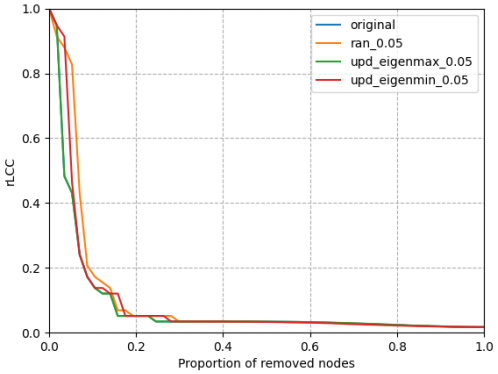
(c) rLCC(non-updated degree(0.05))



(d) rLCC(updated degree(0.05))



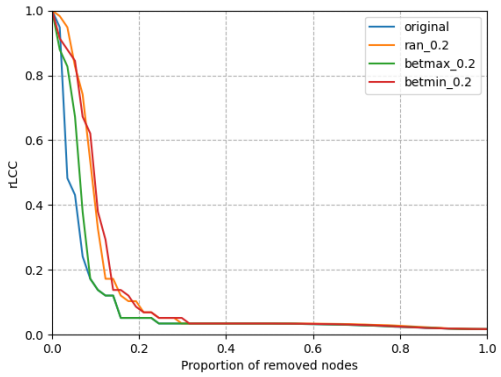
(e) rLCC(non-updated eigenvector centrality(0.05))



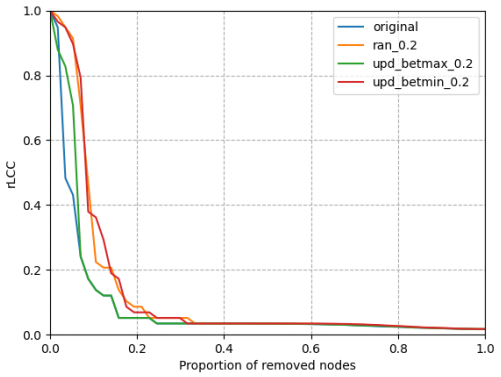
(f) rLCC(updated eigenvector centrality(0.05))

Figure 5.25: Comparison of different link addition strategies on a real-world network named 'Garr201103' under updated betweenness attacks with 5% of links added. The x-axis denotes the proportion of nodes removed, the y-axis denotes the performance metric, rLCC to be specific, which indicates robustness of the network. The blue curves represent the case when no link is added. The orange curves represent the results of random link addition strategies. The green and red curves represent the results of maximum and minimum link addition strategies respectively.

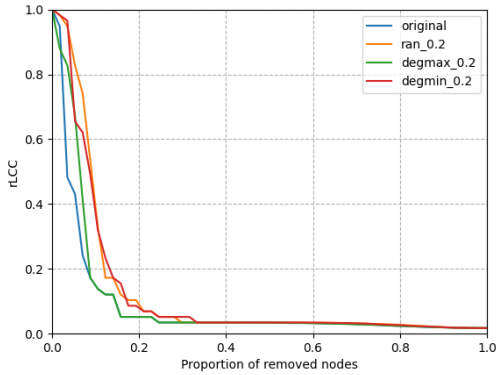
The average rLCC with respect to the proportion of removed nodes for the case of 20% links being added is depicted in Fig. 5.26



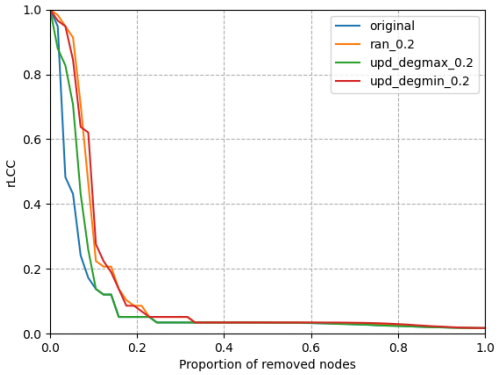
(a) rLCC(non-updated betweenness(0.2))



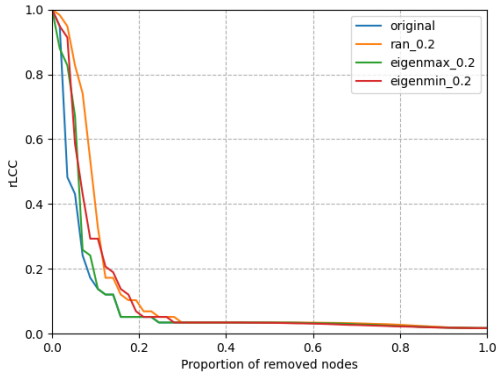
(b) rLCC(updated betweenness(0.2))



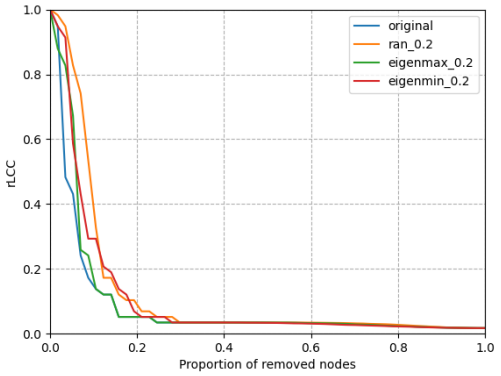
(c) rLCC(non-updated degree(0.2))



(d) rLCC(updated degree(0.2))



(e) rLCC(non-updated eigenvector centrality(0.2))



(f) rLCC(updated eigenvector centrality(0.2))

Figure 5.26: Comparison of different link addition strategies on a real-world network named 'Garr201103' under updated betweenness attacks with 20% of links added. The x-axis denotes the proportion of nodes removed, the y-axis denotes the performance metric, rLCC to be specific, which indicates robustness of the network. The blue curves represent the case when no link is added. The orange curves represent the results of random link addition strategies. The green and red curves represent the results of maximum and minimum link addition strategies respectively.

It is noticed that the impact of protecting strategies becomes more significant

compared with the case when 5% links are added. Random link addition strategy still performs well in this case.

5.2.3. Node protection on synthetic networks

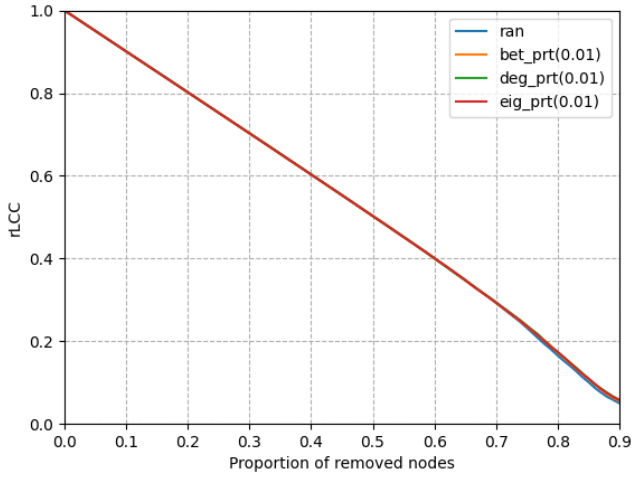
This section assesses the performance of the three node protection strategies. Simulations on an Erdős–Rényi graph are conducted. The property of the network is chosen the same as that in the simulations assessing link addition strategy, with total number of nodes equal to 100 and the edge formation probability equal to 0.1. The performance of different node protection strategies is tested under the scenarios of random attacks and updated betweenness attacks.

Case 1: under random attacks

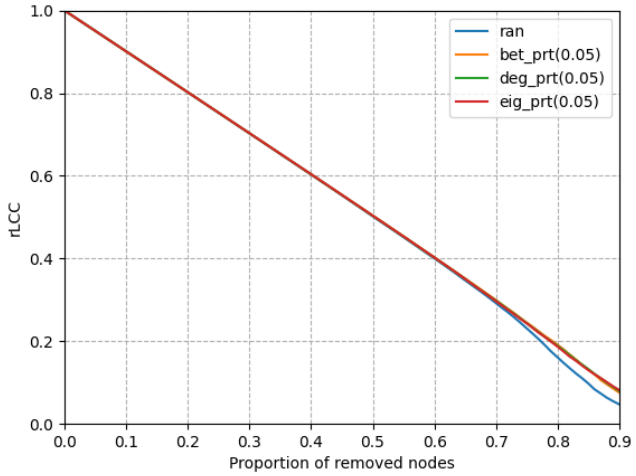
Simulations are conducted on ER random networks G(100, 0.1). In each case, a certain proportion of nodes are protected based on certain properties of the nodes. To provide specific details, the protection strategy involves protecting 1%, 5%, and 10% of the network's nodes with the highest degree, betweenness, and eigenvector centrality. These protected nodes remain intact and are not subject to removal during subsequent ongoing random attacks. The final result for each simulation is obtained by averaging the result of 300 networks after being protected and then attacked. The average rLCC with respect to the proportion of removed nodes is depicted in Fig. 5.29.

The figures indicate that the node protection strategies exert almost no impact before approximately 70% of nodes are removed, and the impact becomes slightly more significant afterwards. To analyse it in detail, when the networks are not protected, the probability for the remaining nodes are removed is $\frac{1}{n-n_{rmv}}$, where n is the total number of nodes and n_{rmv} is the number of removed nodes. When the node protection strategies are applied, the probability for the remaining nodes are removed is $\frac{1}{n-n_{rmv}-n_{prt}}$, where n_{prt} is the number of protected nodes. Consider the case when the networks are best protected (when 10% of the nodes are protected), the probability for the remaining less important nodes are removed becomes $\frac{1}{0.9n-n_{rmv}}$. Compared with the unprotected case, the probability difference to remove the less important nodes is calculated as $\frac{0.1}{(0.9-\frac{n_{rmv}}{n})(1-\frac{n_{rmv}}{n})}$, from which it is found that the difference becomes larger with more nodes removed. That means that with more nodes being removed, the more likely the less important nodes are chosen to be removed when protection strategies are applied. When only a small amount of nodes are removed, the probability difference is small and thus the performance difference is small. However, our observation reveals that despite the protection of the more significant nodes, the robustness of networks experiences only a modest enhancement. It is considered that the ER network already exhibits a degree of robustness under random attacks due to the random nature of edge formation, and protecting 10% of nodes does not exert significant impact on the robustness.

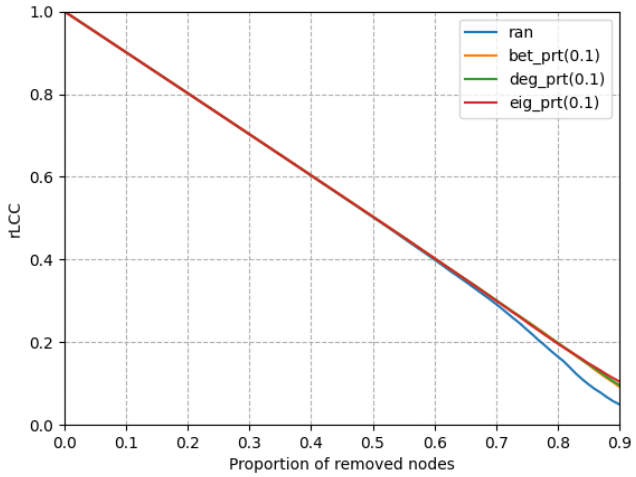
Besides, it is also observed that with more nodes being protected, the networks



(a) rLCC(1%)



(b) rLCC(5%)



(c) rLCC(10%)

Figure 5.27: Protection on Erdős–Rényi networks (100,0.1) under random attacks. The x-axis denotes the proportion of nodes removed, the y-axis denotes the rLCC, which indicates robustness of the network. The blue curves represent the results when no protection is applied. The orange curves represent the results when targeted betweenness-based node protection. The green curves and the red curves represent the results when targeted degree-based and eigenvector centrality-based node protection are applied.

become more robust against attacks.

Case 2: under updated betweenness attacks

Simulations are conducted on ER random networks G(100, 0.1). The protection strategy involves protecting 1%, 5%, and 10% of the network's nodes with the higher degree, betweenness, and eigenvector centrality. These protected nodes remain intact and are not subject to removals during subsequent ongoing updated betweenness attacks. The final result for each simulation is obtained by averaging the result of 300 networks after being protected and then attacked. The average rLCC with respect to the proportion of removed nodes is depicted in Fig. 5.30.

The figures show that the protection strategies perform well on increasing the robustness of the networks under updated betweenness random attacks, especially after more than 40% of nodes are removed. From the plot of 10% of nodes being protected, it is observed that the orange curve, which indicates the targeted betweenness-based node protection, is always above other curves, indicating its best performance among the three protection strategies. This is reasonable, as the networks are under updated betweenness attacks, protecting the nodes with highest betweenness can best maintain the robustness of the networks. The previous observation that with more nodes being protected, the networks become more robust against attacks is also proved according to the figures.

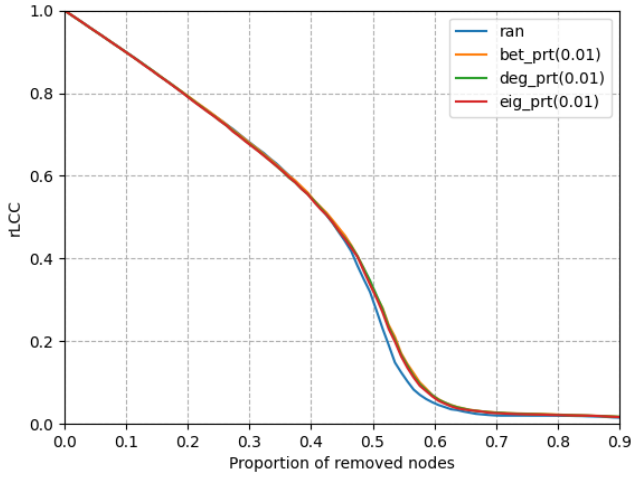
5.2.4. Node protection on real-world networks

This section assesses the performance of the three node protection strategies on real-world networks. Simulations on a network named 'Garr201103' are conducted. The performance of different node protection strategies is tested under the scenarios of random attacks and updated betweenness attacks.

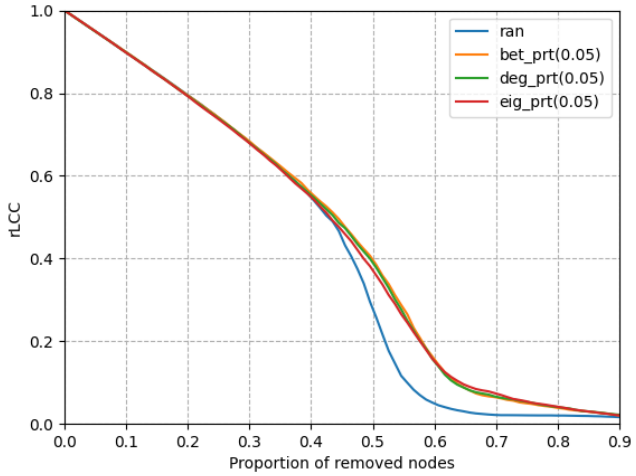
Case 1: under random attacks

Simulations are conducted on the network named 'Garr201103'. The protection strategy involves protecting 1%, 5%, and 10% of the network's nodes with the higher degree, betweenness, and eigenvector centrality. These protected nodes remain intact and are not subject to removal during subsequent ongoing random attacks. The final result for each simulation is obtained by averaging the result of 300 networks after being protected and then attacked. The average rLCC with respect to the proportion of removed nodes is depicted in Fig. 5.29.

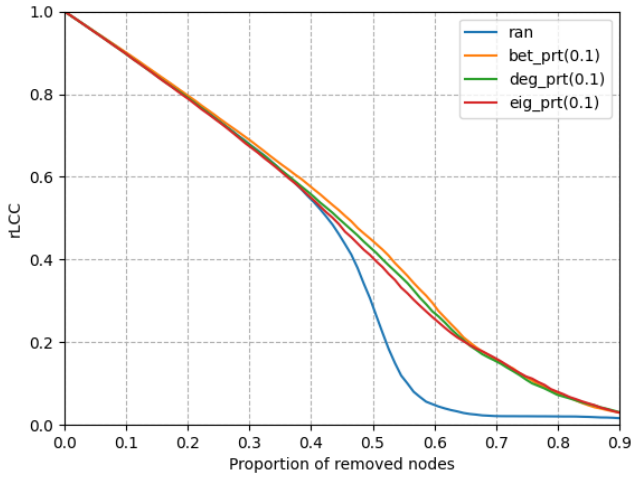
In this case, it is found that the node protection strategies perform well on the real-world network under random attacks, which is different from what is observed in the case of protecting Erdős–Rényi networks. Unlike Erdős–Rényi networks, real-world graphs generally lack robustness against random attacks, and therefore, protecting a certain proportion of critical nodes can significantly increase their robustness.



(a) rLCC(1%)

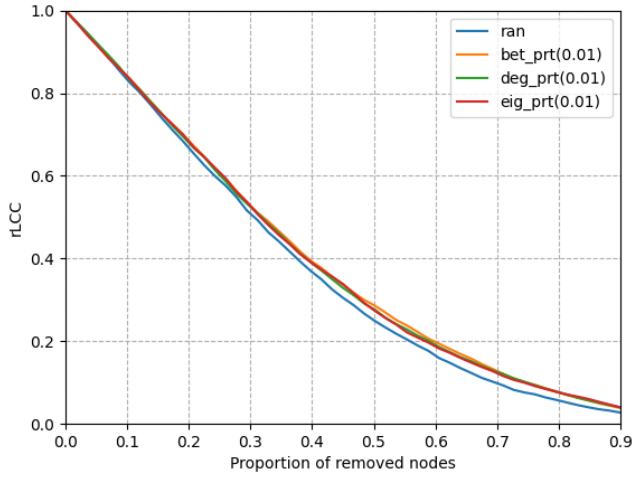


(b) rLCC(5%)

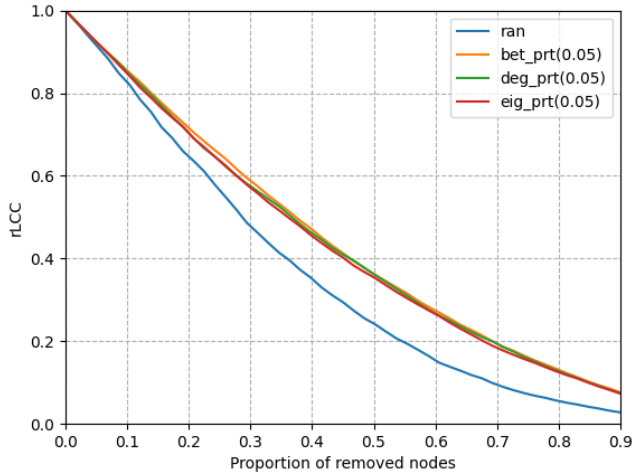


(c) rLCC(10%)

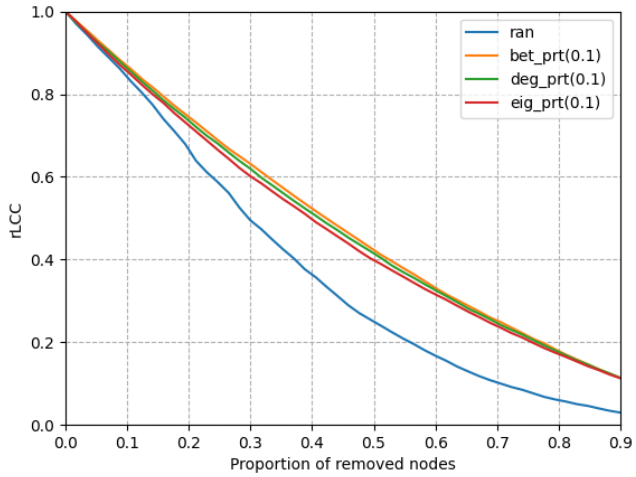
Figure 5.28: Protection on Erdős–Rényi networks (100,0.1) under updated betweenness attacks. The x-axis denotes the proportion of nodes removed, the y-axis denotes the rLCC, which indicates robustness of the network. The blue curves represent the results when no protection is applied. The orange curves represent the results when targeted betweenness-based node protection. The green curves and the red curves represent the results when targeted degree-based and eigenvector centrality-based node protection are applied.



(a) rLCC(1%)



(b) rLCC(5%)



(c) rLCC(10%)

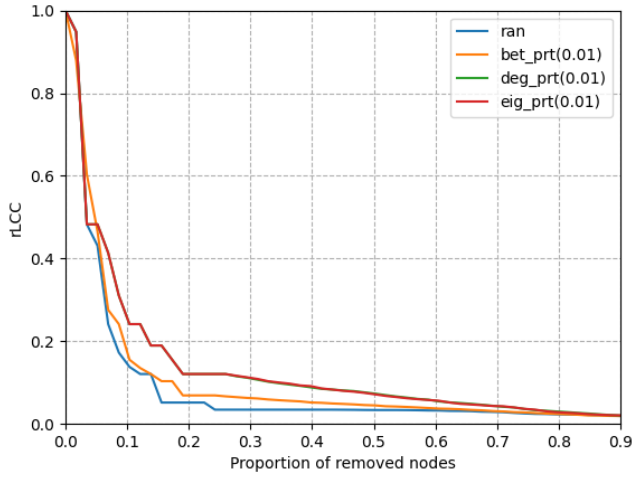
Figure 5.29: Protection on a real-world network named 'Garr201103' under random attacks. The x-axis denotes the proportion of nodes removed, the y-axis denotes the rLCC, which indicates robustness of the network. The blue curves represent the results when no protection is applied. The orange curves represent the results when targeted betweenness-based node protection. The green curves and the red curves represent the results when targeted degree-based and eigenvector centrality-based node protection are applied.

Moreover, it also proves that generally, with more nodes being protected, the networks become more robust against attacks.

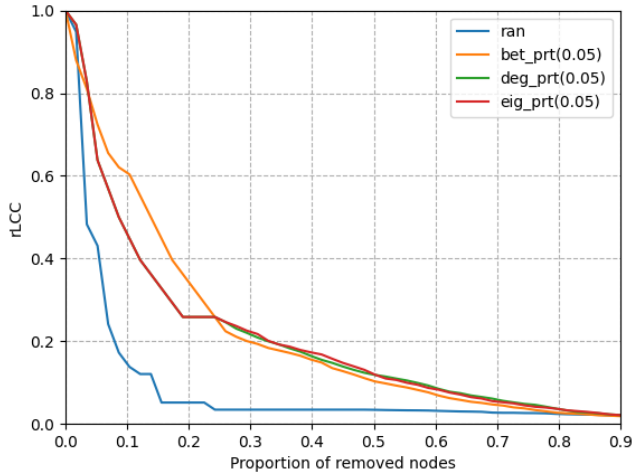
Case 2: under updated betweenness attacks

Simulations are conducted on the network named 'Garr201103'. The protection strategy involves protecting 1%, 5%, and 10% of the network's nodes with the higher degree, betweenness, and eigenvector centrality. These protected nodes remain intact and are not subject to removals during subsequent ongoing updated betweenness attacks. The final result for each simulation is obtained by averaging the result of 300 networks after being protected and then attacked. The average rLCC with respect to the proportion of removed nodes is depicted in Fig. 5.30.

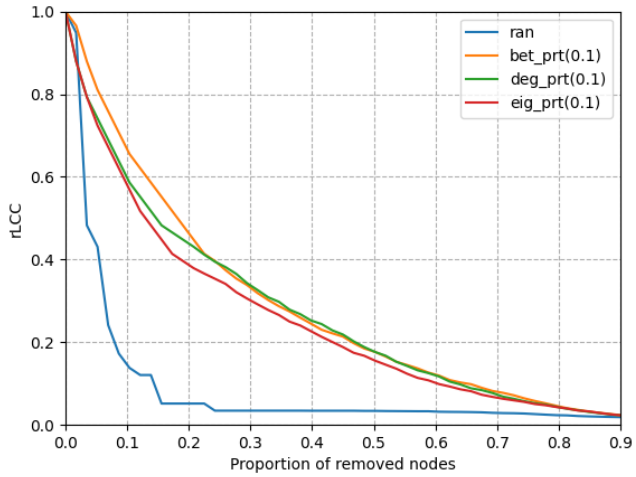
The figures show that the protection strategies perform well on increasing the robustness of the networks under updated betweenness random attacks, even protecting only one node increases the robustness of the networks to some extent. From the plot of 5% and 10% of nodes being protected, it is observed that the orange curves, which indicate the targeted betweenness-based node protection, are above other curves before 25% of nodes removed, indicating its superior performance among the three protection strategies during the initial phase of the attacks. This observation indicates an overall best performance for the targeted betweenness-based node protection among the whole process.



(a) rLCC(1%)



(b) rLCC(5%)



(c) rLCC(10%)

Figure 5.30: Protection on a real-world network named 'Garr201103' under updated betweenness attacks. The x-axis denotes the proportion of nodes removed, the y-axis denotes the rLCC, which indicates robustness of the network. The blue curves represent the results when no protection is applied. The orange curves represent the results when targeted betweenness-based node protection. The green curves and the red curves represent the results when targeted degree-based and eigenvector centrality-based node protection are applied.

6

Conclusion

In this thesis, the research focuses on the evaluation of performance of different approaches applied to assess and increase robustness of networks. It begins with robustness assessment, where ten attack strategies and three analytical approximation methods are implemented and discussed, and a Machine Learning-based method is tested. Afterwards, the increase of robustness is done by comparing the performance of thirteen link addition and four node protection strategies.

For the simulations of ten attack strategies, which includes random attack, targeted degree-based attacks, targeted betweenness-based attacks, and greedy attacks, it is found that the performance of updated betweenness attacks is the best in all simulations on the ER networks, BA networks, and real-world networks, and random attack strategy is always the worst in all the simulations. In the case of synthetic networks, this conclusion is found intuitively by observing the plots, as the curves which indicate updated betweenness attacks and random attack are of the lowest and highest values of rLCC and ATTR respectively in the whole process of attacks. In the case of real-world networks, the conclusion is drawn by calculating the average energy for each attack strategy applied on 233 networks from the Topology Zoo. In the simulations of real-world networks, it is also found that greedy attack strategy always outperforms any other attack strategy at the very beginning, but its performance is surpassed by other attack methods after certain proportion of nodes are attacked. Furthermore, compared with non-updated attack strategies, the updated attack strategies are always better, with the cost of higher computational complexity, as the graph metrics are updated every time after one node is removed.

For the analytical approximations, when they are implemented on ER networks, the approximations of rLCC and ATTR for random attack and stochastic degree attacks show close fit to the simulations, especially in the case of random attack. However, when the approximations are implemented on BA networks, though it shows perfect performance in the case of random attack, the analytical methods for stochastic degree attacks perform badly. When they are applied on real-world networks, even the

performance for the scenario of random attack is notably inadequate. The reason is then explored by conducting simulations on the networks which are of the same degree distribution as the chosen real-world network, and it is concluded that the analytical approximation for random attack is, in the case of random attack, the averaged value of rLCC and ATTR of all configuration models of the same degree distribution.

For the predictions, the SPP-CNN shows good performance in the case of both random and targeted-degree attacks when the targeted predicted models are synthetic networks. Besides, the SPP-CNN also shows acceptable performance in the case of both random and targeted-degree attacks when the targeted predicted models are real-world networks with the prediction error fluctuating around 0.005.

For the protecting strategies, thirteen link addition strategies, including random link addition, targeted degree-based link addition, targeted betweenness-based link addition, and targeted eigenvector centrality-based link addition, are simulated and compared. By computing the energy for each link addition strategy simulation on synthetic network, it is found that the updated minimum betweenness link addition strategy always demonstrates superior network protection performance with highest value of energy always observed, and thus it is considered that the updated minimum certain metric link addition strategy is the best when the networks are under corresponding attacks. In addition, the maximum certain metric link addition strategies perform much worse than the minimum certain metric link addition strategies, and the updated strategies outperform the non-updated strategies. In the case of real-world networks, one more observation is found that random link addition strategy shows exceptional performance in the simulations. In terms of node protection, it is analysed that the targeted certain metric-based node protection strategy is the best when the networks are under corresponding attacks.

For future work, we would like to address the following aspects:

1. Focusing on the topological properties of real-world networks, we should find out the relationship between them and the performance of attacks and protecting strategies.
2. We should further investigate the analytical approximation approaches, especially the approximations for targeted degree-based attacks, exploring the reason behind the inadequate performance on both synthetic and real-world networks.
3. We should optimize the machine learning model, or find a more suitable model to predict the robustness of networks, with that the existing model shows not perfect performance on real-world networks. The optimization can also focus on the training data set. By analyzing the property of the networks in the training data set, we should find out the most proper data set to be trained so that the prediction error is minimised.

References

[1] Mark Newman. *Networks: An Introduction*. Oxford University Press, Mar. 2010.

[2] Kon Shing Kenneth Chung, Mahendra Piraveenan, and Liaquat Hossain. "Topology of Online Social Networks". In: *Encyclopedia of Social Network Analysis and Mining*. Ed. by Reda Alhajj and Jon Rokne. New York, NY: Springer New York, 2014, pp. 2191–2202.

[3] Cunlai Pu and Pang Wu. *Vulnerability Assessment of Power Grids Based on Both Topological and Electrical Properties*. Sept. 2019.

[4] P Van Mieghem et al. "A framework for computing topological network robustness". In: *Delft University of Technology, Report20101218* (2010), pp. 1–15.

[5] Béla Bollobás. *The Evolution of Random Graphs—the Giant Component*. 2nd ed. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2001, pp. 130–159.

[6] Sebastian Neumayer et al. "Assessing the Vulnerability of the Fiber Infrastructure to Disasters". In: *IEEE/ACM Transactions on Networking* 19.6 (2011), pp. 1610–1623.

[7] Christian Schneider et al. "Mitigation of malicious attacks on networks". In: *Proceedings of the National Academy of Sciences of the United States of America* 108 (Feb. 2011), pp. 3838–41.

[8] Cun-Lai Pu, Wen-Jiang Pei, and Andrew Michaelson. "Robustness analysis of network controllability". In: *Physica A: Statistical Mechanics and its Applications* 391.18 (2012), pp. 4420–4425.

[9] Zhe-Ming Lu and Xin-Feng Li. "Attack Vulnerability of Network Controllability". In: *PLOS ONE* 11.9 (Sept. 2016), pp. 1–27.

[10] M. E. J. Newman, S. H. Strogatz, and D. J. Watts. "Random graphs with arbitrary degree distributions and their applications". In: *Phys. Rev. E* 64 (2 July 2001), p. 026118.

[11] Dror Y. Kenett et al. *Network of Interdependent Networks: Overview of Theory and Applications*. Ed. by Gregorio D'Agostino and Antonio Scala. Cham: Springer International Publishing, 2014, pp. 3–36.

[12] Fenghua Wang and Robert E. Kooij. "Robustness of Network Controllability with Respect to Node Removals Based on In-Degree and Out-Degree". In: *Entropy* 25.4 (2023).

[13] Yang Lou et al. "Knowledge-Based Prediction of Network Controllability Robustness". In: *IEEE Transactions on Neural Networks and Learning Systems* 33.10 (2022), pp. 5739–5750.

[14] Yang Lou et al. “A Learning Convolutional Neural Network Approach for Network Robustness Prediction”. In: *IEEE Transactions on Cybernetics* 53.7 (July 2023), pp. 4531–4544.

[15] Chengpei Wu et al. “SPP-CNN: An Efficient Framework for Network Robustness Prediction”. In: *arXiv preprint arXiv:2305.07872* (May 2023).

[16] Yang Lou et al. “Network Robustness Prediction: Influence of Training Data Distributions”. In: *IEEE Transactions on Neural Networks and Learning Systems* (2023), pp. 1–12.

[17] Huijuan Wang and Piet Van Mieghem. “Algebraic Connectivity Optimization via Link Addition”. In: *Proceedings of the 3rd International Conference on Bio-Inspired Models of Network, Information and Computing Sytems*. Brussels, BEL: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.

[18] V. H. P. Louzada et al. “Smart rewiring for network robustness”. In: *Journal of Complex Networks* 1.2 (Sept. 2013), pp. 150–159.

[19] Piet Van Mieghem. *Performance Analysis of Complex Networks and Systems*. Cambridge University Press, 2014.

[20] Piet van Mieghem. *Spectra of complex networks*. Cambridge University Press, 2010, pp. 179–208.

[21] Javier Martın Hernández and Piet Van Mieghem. “Classification of graph metrics”. In: *Delft University of Technology, Report20111111* (2011), pp. 1–20.

[22] Paul Erdős, Alfréd Rényi, et al. “On the evolution of random graphs”. In: *Publ. Math. Inst. Hung. Acad. Sci 5.1* 5.1 (1960), pp. 17–60.

[23] Albert-László Barabási and Réka Albert. “Emergence of scaling in random networks”. In: *Science* 286.5439 (1999), pp. 509–512.

[24] Mark Newman. *The configuration model*. Oxford University Press, July 2018.

[25] Simon Knight et al. “The Internet Topology Zoo”. In: *IEEE Journal on Selected Areas in Communications 29.9* (2011), pp. 1765–1775.

[26] Rong Zhou Nesreen Ahmed Ryan A. Rossi. *Network Repository. An Interactive Scientific Network Data Repository.* `https://networkrepository.com/social-network-2023`.

[27] David P. Williamson and David B. Shmoys. *The Design of Approximation Algorithms.* Cambridge University Press, 2011, pp. I–XI, 1–504.

[28] Herbert S. Wilf. *Generatingfunctionology*. USA: A. K. Peters, Ltd., 2006.

[29] Mark EJ Newman. “The structure and function of complex networks”. In: *SIAM review* 45.2 (2003), pp. 167–256.

[30] Jia Shao et al. “Structure of shells in complex networks”. In: *Physical Review E* 80.3 (2009), p. 036105.

[31] “Errors of prediction and least-squares estimation”. In: *Understanding Regression Analysis*. Boston, MA: Springer US, 1997, pp. 21–25.