University of Zurich[UZH]

MASTER THESIS – Communication Systems Group, Prof. Dr. Burkhard Stiller

An Architectural Evaluation of Future Internet Architectures and Their Censorship Resilience

Linn Anna Spitz
Zurich, Switzerland
Student ID: 17-923-665

Supervisor: Thomas Grübl, Dr. Francesca Falzon, Prof. Dr. Burkhard Stiller
Date of Submission: April 19, 2026

University of Zurich
Department of Informatics (IFI)
Binzmühlestrasse 14, CH-8050 Zürich, Switzerland

ifi

Master Thesis
Communication Systems Group (CSG)
Department of Informatics (IFI)
University of Zurich
Binzmühlestrasse 14, CH-8050 Zürich, Switzerland
URL: http://www.csg.uzh.ch/

Declaration of Independence

I hereby declare that I have composed this work independently and without the use of any aids other than those declared (including generative AI such as ChatGPT). I am aware that I take full responsibility for the scientific character of the submitted text myself, even if AI aids were used and declared (after written confirmation by the supervising professor). All passages taken verbatim or in sense from published or unpublished writings are identified as such. The work has not yet been submitted in the same or similar form or in excerpts as part of another examination.

Zürich, April 17, 2026

Signature of student

Abstract

In today's Internet landscape, censorship is a pressing subject, and network topologies play a crucial role in determining censorship resilience. Future Internet Architectures (FIA) propose alternatives to the current model based on the Border Gateway Protocol (BGP) and the Internet Protocol (IP), but often do not address censorship resilience as a primary design goal, resulting in a lack of systematic evaluation and optimization methodologies for such resilience. SCION, an FIA for inter-domain routing, introduces Isolation Domains (ISD) and multi-path routing, which induce topological properties that impact censorship resilience.

To address the research gap, this work presents a theoretical topological analysis of multiple Internet architectures, including SCION. It further provides an extensive survey of graph-theoretic robustness metrics applicable to censorship resilience evaluation and proposes a taxonomy of these metrics. Additionally, Border Breadth, a SCION-specific resilience metric, is introduced. It quantifies the structural influence of core nodes within an ISD relative to its size.

Two edge-rewiring algorithms, the adapted R_{AC} and the proposed R_{NP}, are developed to optimize resilience in SCION topologies. To enable experimentation on operational networks, a configurable SCION testbed was implemented. The empirical results, including evaluations on the real-world SCIERA network, show that both algorithms significantly improve censorship resilience while preserving the number of edges, with R_{AC} outperforming R_{NP} overall.

Kurzfassung

In der heutigen Internetlandschaft ist Zensur ein drängendes Thema, wobei die Netzwerktopologie eine entscheidende Rolle spielt. Future Internet Architectures (FIA) schlagen Alternativen zum aktuellen Modell vor, welche auf dem Border Gateway Protocol (BGP) und dem Internet Protocol (IP) basieren. FIAs sind jedoch häufig nicht primär auf Zensurresilienz ausgerichtet, was zu einem Mangel an systematischen Auswertungen und Optimierungsmethoden führt. SCION, ein FIA für Inter-Domain-Routing, führt Isolation Domains (ISD) und Multipath-Routing ein, die topologische Eigenschaften hervorbringen und die Zensurresilienz beeinflussen.

Um die Forschungslücke zu adressieren, präsentiert diese Arbeit eine theoretische Analyse mehrerer Internetarchitekturen, einschliesslich SCION. Des Weiteren liefert sie eine ausführliche Übersicht über bestehende Robustheitsmetriken und schlägt eine Taxonomie der Metriken vor. Zusätzlich wird die Border Breadth, eine SCION-spezifische Resilienzmetrik, eingeführt. Sie quantifiziert den strukturellen Einfluss der Core-Knoten innerhalb eines ISD im Verhältnis zu dessen Grösse.

Zwei Algorithmen zur Kanten-Neuverdrahtung, der angepasste R_{AC} und der neuartige R_{NP}, werden zur Resilienzoptimierung von SCION-Topologien entwickelt. Um Experimente auf operativen Netzwerken zu ermöglichen, wurde eine konfigurierbare SCION-Testumgebung implementiert. Die empirischen Ergebnisse, einschliesslich Auswertungen auf dem realen SCIERA-Netzwerk, zeigen, dass beide Algorithmen signifikante Verbesserungen der Zensurresilienz erzielen, wobei R_{AC} insgesamt bessere Ergebnisse liefert.

Acknowledgments

I would like to sincerely express my gratitude to my supervisors, Thomas Grübl and Dr. Francesca Falzon. Their valuable feedback, support, and generous time commitment have been crucial for my thesis.

Additionally, I am grateful to Prof. Dr. Burkhard Stiller for the opportunity to complete my Master thesis at the Communication Systems Group (CSG) of the University of Zurich. The process has been a valuable learning experience through which I have gained much insight and new scientific skills. Finally, I would like to thank Stefan Kraft for proof-reading my thesis.

Contents

Chapter 1

Introduction

Future Internet Architectures (FIA) are research efforts that fundamentally aim to address perceived flaws in the standard model based on the Border Gateway Protocol (BGP) and the Internet Protocol (IP) by proposing alternative design solutions. One such architecture is the SCION network, which addresses inter-domain routing between Autonomous Systems (AS) and therefore constitutes a substitute for the standard BGP architecture. Further, in today's Internet landscape, censorship and surveillance have become a pressing subject. These dimensions should be carefully considered in the domain of FIAs. Therefore, this work conducts an examination of the relationship between censorship resilience and the SCION architecture.

1.1 Motivation

The discussion about censorship and surveillance has become increasingly relevant since the inception of the Internet. A 2025 study by Freedom House reported a global decrease of freedom of information on the Internet for the past 15 consecutive years [1]. It is therefore central to consider which architectural features of the Internet are conducive to freedom of information and which introduce vulnerabilities.

Prior literature shows that topological aspects of the Internet are an important factor for censorship resilience. It has been repeatedly confirmed that bottleneck structures render a topology more vulnerable to the restriction of information [2], [3], [4], [5].

Many FIAs address issues in the current Internet architectures related to security. While some previous work examines the censorship resilience of FIAs, it is often not treated as a primary design objective [6].

The SCION architecture introduces two important factors on a topological level. Firstly, it partitions the global AS network into subgroups called Isolation Domains (ISD), where only a subset of nodes handle inter-ISD communications. Further, SCION includes full multi-path capabilities. It is the position of this work that these two factors introduce risks and opportunities for SCION with regard to censorship. The partitioning of ISDs

could potentially result in greater control over information flow. On the other hand, multi-path routing mitigates concentrated points of control.

With these factors established, this work investigates the implications of the SCION topology on censorship resilience.

1.2 Description of Work

Building on previous work conducted in [7], this work further analyzes SCION and its architectural impacts on censorship resilience. Path diversity is used as a fundamental metric for censorship resilience. Further, the analysis is built on the equivalence between censorship resilience and failure robustness on a graph theoretical level. Failure robustness is a well-studied subject in graph theory and network science. Therefore, a vast body of robustness metrics exists. This work includes a survey of relevant robustness metrics and proposes a novel taxonomy. Furthermore, the metric *Border Breadth* is introduced, which is a novel metric specific to SCION. It quantifies the relative structural influence of core nodes within an ISD relative to its size.

In a subsequent step, this paper proposes a process to improve the censorship resilience of a given SCION topology. This process aims to optimally leverage SCION's multi-path capabilities by increasing path diversity through minimal modifications to the network topology. Two algorithms R_{AC} and R_{NP} are applied with the aim of producing such favorable modifications. The topological alterations are limited to edge rewirings, hence keeping the number of nodes and edges in the network constant.

In order to evaluate the impact of different topologies on the SCION architecture, this work includes a fully configurable SCION testbed, where AS nodes are simulated by Docker containers. Building on a pre-existing implementation, a modified architecture allows for fully configurable topologies and includes tools for measurements and evaluation.

Using the configurable testbed, a thorough evaluation of the optimization processes is conducted. Using 12 baseline topologies, both optimization algorithms are applied. Each iteration is recorded, resulting in 132 topologies. The analysis includes both theoretical robustness metrics as well as the experimental deployment of the topologies on the testbed, which allows for empirical measurements. Among the 12 baseline topologies is the SCIERA network [8], a fully functional SCION network connecting research institutions globally. This provides a real-world example of a SCION network and how it could potentially be optimized.

Chapter 2

Background

This research investigates the censorship resilience of Future Internet Architectures from a topological perspective. In this chapter, the most relevant background information from three distinct fields is summarized, whereby the research motivation is further illustrated simultaneously. First, the context of Internet censorship and surveillance is established. Next, prominent examples of FIA are introduced, with emphasis on the SCION architecture. Finally, a brief section introduces basic graph theory concepts relevant to the topological analyses, as well as the notation used throughout this work.

2.1 Internet Censorship and Surveillance

As the global Internet became the world's central means of communication, Internet censorship emerged with it. In recent years, the urgency of the matter has only increased as the Internet has permeated almost every aspect of daily life. Disconcertingly, Freedom House reported a decrease in global Internet freedom in 2025, following a trend of 15 consecutive years [1]. Freedom House lists Myanmar, China, Iran, Russia, and Belarus as the least free countries with regard to the Internet. Additionally, in 2013, Edward Snowden revealed the massive extent of the surveillance apparatus employed by the Five Eyes nations, which consist of the US, the UK, Australia, Canada, and New Zealand [9]. Furthermore, the European Union monitors networks for accordance with its legislation. Evidence suggests that EU member states block more websites than indicated on public block lists [10]. Overall, it has become increasingly clear that governments across the globe have an interest in controlling the flow of information.

Having established the ubiquity of the issue, the following subsection discusses the most common methods of Internet censorship and surveillance.

2.1.1 Censorship and Surveillance Methods

There are various ways to control access to online information. In [11], the authors measure 70 countries with respect to the censorship they employ. Web-based protocols such as Transport Layer Security (TLS), Domain Name System (DNS) and Hypertext Transfer Protocol (HTTP), as well as identifiers like Uniform Resource Locator (URL), are often targeted by censors. The censorship methods evaluated are (1) Internet shutdowns, (2) IP address or port blocking, (3) BGP attacks and disruption, (4) bandwidth throttling, (4) DNS tampering, (5) HTTP/URL/keyword filtering, (6) TLS-based filtering, and (7) protocol fingerprinting. Their results suggest that HTTP/URL/keyword filtering and TLS-based filtering are among the most prominent methods.

Furthermore, in an effort to systematize censorship, the authors of [12] present a taxonomy of censorship methods. They propose that methods can be categorized by attack mode among other dimensions. The attack mode comprises the source of interest and an associated action. The authors identify nodes, links, and users as sources of interest. Attacks on nodes include Denial-of-Service (DoS) attacks and server take-downs, whereas attacks on links include IP blocking and filtering, DNS tampering, and HTTP filtering, and user attacks include surveillance methods. Referring back to the results found in [11], this implies that link attacks account for the most prevalent censorship methods.

Furthermore, it should be noted that censorship methods are not exclusively technical in nature. According to [13], censorship research often focuses on authoritarian regimes. However, democracies also employ censorship, and often do so through private actors, like internet service providers or online content providers. Policy decisions play a major role in censorship [9], however this work focuses on its technical aspects.

To further understand the mechanisms of censorship, the authors of [14] probe the topology of the Iranian internet and identify that all sampled requests are censored by a single node through which all traffic had been routed. This implies that censoring entities rely on a heavily centralized topology to enforce their policies. Other works have also examined the relationship between censorship resilience and AS topology [2], [4], [5], [7]. In [4], the authors evaluate national AS-level chokepoints and find a negative correlation between national choking potential and freedom on the internet and press freedom. This work focuses on the topological aspects of censorship, specifically in the context of the SCION architecture. A survey on the relation of internet topology and censorship is presented in Chapter 3. The next section provides an overview of prominent FIAs.

2.2 Future Internet Architectures

Future Internet Architectures are a set of proposals that aim to replace certain aspects of the current global Internet. Many have pointed out flaws in both the current Transmission Control Protocol/Internet Protocol (TCP/IP) design, as well as in the BGP architecture. Certain legacy design decisions are based on the Internet at the time of its inception, where known, trusted entities stood in mutual exchange. Consequently, as the need for security features rose, they were retrofitted to the existing design [6]. In response to the

limitations of the original internet design, several initiatives were launched that aim to rethink important aspects of networking.

2.2.1 NSF Architectures

To address the challenges posed by the current internet architecture, the US National Science Foundation (NSF) launched a Future Internet Architecture program. The following paragraphs list and briefly discuss the FIAs funded by the program.

Nebula

The Nebula architecture aims to address the rising demand for high-performance secure cloud computing [15]. It consists of the Nebula core, which is a set of ultra-reliable high performance routers that form the backbone of the network. Furthermore, it has an extensible control plane (NVENT) and a data plane (NDP) that allows for communication only when all parties, including hosts, users, and intermediary nodes, have consented. To enforce this, it uses a path verification mechanism called ICING.

Named Data Network

Named Data Networking (NDN) aims to shift from the current host-centric paradigm to a content-centric one [16]. In this architecture, users request specific content rather than a network address. The network resolves this request by identifying and delivering the closest location of a given piece of content. NDN uses unique human-readable names to identify content. Similar to a file path, names are split by forward slashes into a hierarchical description. NDN differentiates between two types of packets, interest packets and data packets, analogous to the request-response paradigm. In contrast to the conventional Internet architecture, NDN routers announce name prefixes, rather than address prefixes, to their neighbors. This information is propagated through the network. The routers store a Pending Interest Table (PIT), which caches unanswered interest packets, a Forwarding Information Base (FIB), which caches name prefixes of other routers, and the Content Store (CS), storing the content it serves.

MobilityFirst

As the name suggests, MobilityFirst aims to design an internet where mobile devices are presumed to be the primary constituents of traffic [17]. One of its fundamental propositions are the *principals*, which comprise devices, files, services, human end-users, etc. The architecture separates principals, which are static, from networks, which are dynamic. In this manner, MobilityFirst addresses the mobile design principle, where the same mobile devices frequently change networks. Principals and networks are identified by Global Unique Identifiers (GUID) and Network Addresses (NA), respectively. The Global Name Service (GNS) is of central importance, as it stores a GUID's current NA.

Expressive Internet Architecture

The Expressive Internet Architecture (XIA) focuses on interoperability and evolvability [18]. Similarly to MobilityFirst, it also defines principals, which can be content, services, users, etc. XIA is built on the assumption that new types of principals are likely to emerge. It provides a mechanism using *fallbacks* to seamlessly support the incremental adoption of new technology. Fallbacks specify a router's alternative actions if the primary behavior is somehow impeded. Furthermore, XIA aims to be intrinsically secure, and requires all principals to be authenticated.

2.2.2 SCION

SCION (Scalability, Control and Isolation On next-generation Networks) is a Future Internet Architecture designed as an alternative to current inter-domain routing protocols [19]. Currently, inter-domain routing is largely based on BGP [20], which handles connections between Autonomous Systems. SCION aims to address the following limitations in the current system.

(1) Routing information announced by ASes can propagate arbitrarily through the network, potentially resulting in security risks. Given the global nature of routing updates in BGP, this also results in stale information, as change propagates incrementally.

(2) BGP determines routes on a hop-to-hop basis. It usually only provides a single path per prefix and neither the source nor the destination has control over the path. Traffic routed from A to B might therefore traverse intermediaries which the endpoints deem untrustworthy.

SCION introduces the following properties to address these drawbacks.

(1) *Isolation Domains* are subsets of ASes usually linked through shared characteristics such as geographic location or economic relationships. Routing updates are contained within Isolation Domains, which increases security in the network.

(2) SCION offers *path transparency*, which allows end-hosts to verify the path a packet has traversed. Furthermore, SCION aims for complete *path control* for end-hosts. In this case, receivers select a set of paths leading to their address and senders select which path they want to use for a packet.

A real-world example of a SCION deployment is SCION Education, Research, and Academic (SCIERA), a network infrastructure interconnecting academic institutions worldwide [8]. The objective of this initiative is to foster SCION-native applications that are fully SCION-aware and therefore able to leverage its capabilities. The association publishes an overview of the current network topology [21]. The network currently comprises 29 autonomous systems (ASes).

Isolation Domains

ISDs are a logical grouping of ASes, for example according to a common jurisdiction or shared economic contracts. They isolate a trust within the domain. Routing information therefore does not have to be shared globally across the internet, but can stay securely within one domain. Consequently, there are two types of routing in SCION: intra-domain and inter-domain. Each ISD has a core of highly connected nodes. These serve both as important relay nodes for intra-domain routing as well as gateway routers for inter-domain routing. The cores of different ISDs are well connected among each other. Figure 2.1 illustrates ISDs, their cores, and their interconnectedness.

Furthermore, ASes can be part of several ISDs. Therefore, ISDs can overlap. Inter-domain links among non-core nodes are also possible, as illustrated in Figure 2.1.

Each ISD core establishes a Trust Root Configuration (TRC), which allows for verification of name bindings and public keys in the ISD. When an AS joins the ISD, it agrees to the ISD's TRC.

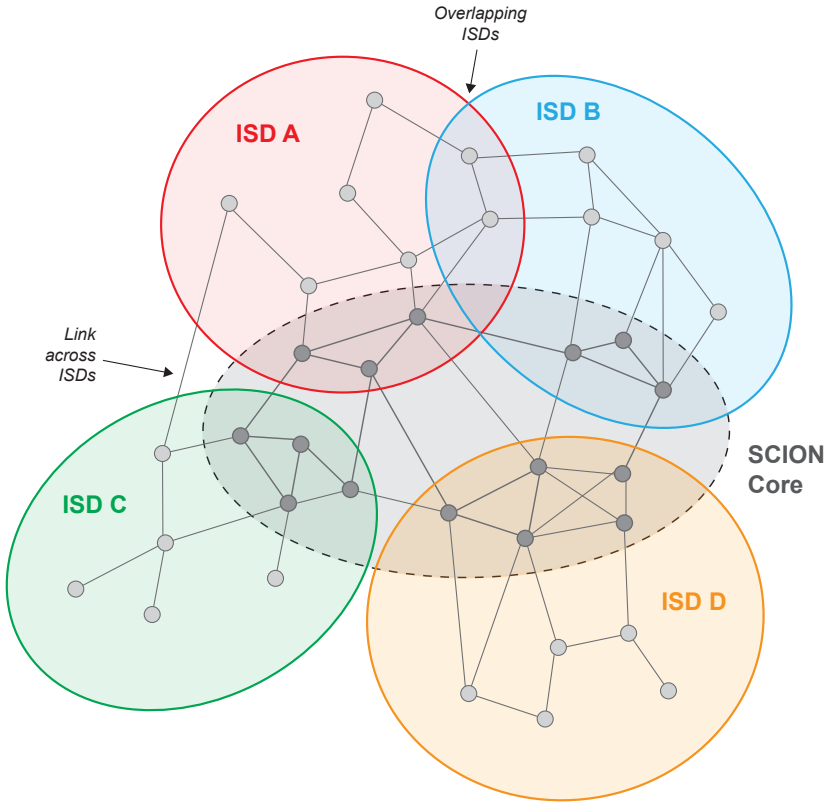

Figure 2.1: The SCION architecture with 4 ISDs, adapted from [19].

2.3 Graph Theory Concepts

Graph theory is a broad field concerned with the study of graphs and complex networks. This section briefly introduces the concepts most relevant to this work, as well as the notation used henceforth.

Fundamentally, a graph is a data structure consisting of vertices V and edges E that form graph G, with nodes $i = 1, ..., n$ and edges (i, j). A graph is often denoted as a tuple $G = (V, E)$. The number of neighbors of a node i is known as its degree, $\deg(i)$.

The topology of a graph can be described with an adjacency matrix A of size $n \times n$. The adjacency matrix is defined as follows:

$$A_{ij} = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$$

Common types of graphs include undirected, directed, weighted, unweighted, and simple graphs. In a weighted graph, the entries of A correspond to the edge weights. If a graph is undirected, the adjacency matrix is symmetric. Graph representations in this work are limited to simple graphs, which are undirected, unweighted, and contain no self-loops, so that $A_{ii} = 0$ for all nodes i.

Given a graph G, a subgraph S is defined by a subset of vertices $V(S) \subseteq V$ and the edges $E(S) \subseteq E$ that internally connect them, where $\forall (u, v) \in E(S) \implies u \in V(S) \wedge v \in V(S)$. Furthermore, graph subtraction is denoted as $Q = G - S$, with the resulting subgraph Q containing the nodes $V(Q) = V - V(Q)$ and the edges as $E(Q) = E - \{(u, v) \in E : u \in V(S) \vee v \in V(S)\}$. Finally, the set of edges that connect the disjoint subgraphs S and Q is denoted as $\partial(S)$, so that $\forall (u, v) \in \partial(S) \implies (u \in V(S) \wedge v \in V(Q)) \vee (u \in V(Q) \wedge v \in V(S))$, and with $\partial(S) = \partial(Q)$.

Many graph metrics are based on the eigenvalues and eigenvectors of the adjacency matrix A. The set of eigenvalues of A is denoted as $\{\lambda_i\}_{i \in [n]}$, with $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_n$. The corresponding set of eigenvectors is denoted as $\{\mathbf{u}_i\}_{i \in [n]}$.

The Laplacian matrix is an additional structure that is used to describe graphs. It is defined as follows:

$$L_{ij} = \begin{cases} \deg(i) & \text{if } i = j \\ -1 & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$$

The set of eigenvalues of L is denoted as $\{\mu_i\}_{i \in [n]}$, with $\mu_1 \leq \mu_2 \leq ... \leq \mu_n$, and corresponding eigenvectors $\{\mathbf{v}_i\}_{i \in [n]}$. Note that, in contrast to the adjacency matrix, eigenvalues are sorted in ascending order and $\mu_1 = 0$ for all undirected graphs.

This work focuses on graph metrics applicable to censorship resilience and robustness of a graph. Relevant metrics are surveyed, categorized, and discussed in Chapter 4.

Chapter 3

Related Work

This work evaluates the censorship resilience of the SCION architecture in a quantitative, topological manner, which spans several areas of research. This chapter summarizes the relevant literature in each area using a semi-systematic review approach. First, prior work on censorship and surveillance within the domain of FIAs is discussed. Secondly, previous studies concerned with topological aspects of censorship are examined. Finally, a survey of previous research on rewiring methods for network robustness optimization concludes this chapter. Each review was conducted using a Google Scholar keyword search, as well as related articles and citation referrals.

3.1 Censorship and Surveillance in FIAs

Given the future-oriented nature of FIAs, this review covers research from approximately the past 10 years. Despite the interest in FIAs, few works specifically address the aspect of censorship in FIAs. The aim of this review is to gather insight on which architectural aspects of FIAs may potentially strengthen or weaken censorship resilience.

In [22], the authors conduct a security and privacy analysis of FIAs, specifically projects funded by the US National Science Foundation. FIA security features are analyzed with regards to the following necessary properties: trust, origin and peer authentication, data integrity, access control, accountability, data and traffic confidentiality, anonymous communication, and availability. The authors analyze each FIA qualitatively, discussing the security features available in each architecture. In a similar fashion, Ding et al. survey the security aspects of FIAs [23]. Among other observations, they point out that the semantic link between names and content poses privacy threats in content-based architectures. Furthermore, the routers' data caches are a natural point of attack. The authors also discuss the additional security features within SCION's architecture. Highly relevant in the context of surveillance is the HORNET system [24], a high-performance onion routing system built for certain FIAs, including SCION. Overall, the authors of [23] find that SCION achieves a higher security and privacy level than other FIAs.

Wrana et al. [6] question the effect of FIAs on Internet censorship. Similarly to works mentioned previously, they conduct a qualitative analysis of six FIAs, including SCION, and analyze their vulnerabilities with regard to censorship and surveillance. The authors posit that, while security has been a focus in the design and development of many FIAs, privacy has been largely overlooked. Threats considered by the study include packet inspection, name resolution manipulation, traffic analysis, packet manipulation, and network caching infrastructure attacks. The latter is only applicable to content-centric architectures such as NDN and XIA, where it is, however, likely to constitute a key point of attack, as others have pointed out [23]. Furthermore, as FIAs rely on mechanisms similar to current name resolution, the authors conclude that DNS based censorship would still be applicable. Additionally, as more complex routing protocols contain more header information, the risk of traffic analysis might increase. With regards to SCION specifically, the authors point to the possibility of manipulating the SCION path construction process. An adversary can achieve this by blocking the communication between two ASes under its control, thereby forcing traffic to take a different, potentially compromised route. Furthermore, they identify a risk posed by the organization of the Internet into ISDs, as this might afford actors such as nation-states more granular control over Internet traffic.

The previous examples exclusively employ qualitative approaches. In [7], the authors develop and apply a quantitative method to compare the censorship resilience of three different Internet architectures on the inter-domain level. The paper compares the BGP model, the waypoint model, which extends the BGP through intermediate nodes such as VPNs, and the SCION architecture. The authors establish a metric called *Censorship Resilience Potential* (CRP). Given a a set of censoring border nodes C, a set of source nodes S and destination nodes D, the CRP measures the fraction of non-intercepted connections between S and D. This metric is applied to different countries, where S and D represent internal and external ASes, respectively. The calculations are carried out using data provided by the Center for Applied Internet Data Analysis (CAIDA), which provides datasets on AS relations and AS information. As the SCION architecture, while already deployed, is still largely prospective on an international level, a realistic SCION dataset is generated as follows. Using the same AS relation data as for the BGP simulation, the global network of ASes is split into ISDs based on countries. The ISD cores are defined as the set of ASes with the largest customer beacon size. As the cores form the connection between ISDs, they also take on the role of border ASes for their countries. Evaluating the CRP across countries and architectures, the results show that the waypoint model and the SCION architecture largely outperform the classical BGP, likely due to their path-aware nature.

The previous work summarized here shows that FIAs present both risks and opportunities with regard to censorship. No work other than [7] has been identified that examines FIAs with regard to censorship in a quantitative, topological manner. However, the same study conducts evaluations on a topological model, where censoring nodes are defined *a priori* and the metrics are calculated *a posteriori*. Therefore, the applied methodology is an experimental approach, simulating possible scenarios. In this research, the theoretical evaluation of the topology is conducted using an analytical method, using metrics which are independent of a predefined set of censoring nodes. In the next section, previous research concerned with the topological aspects of Internet censorship is discussed.

3.2 AS Topology and Censorship

In contrast to censorship in FIAs, there exists a substantial body of research on censorship more broadly. While censorship research is concerned with many different aspects of the Internet, network topology has been identified as a key factor in numerous studies [25]. This review covers research on the relationship between topology and censorship from the past 15 years, limited to publications concerned with AS topology and inter-domain routing.

While it is well-known that China censors much of its Internet traffic, the authors of [5] investigate where on the AS level the filtering occurs. They draw a distinction between border ASes, which peer with foreign ASes, and internal ASes. The study finds a total of 138 internal and 24 border ASes. Two border ASes (the ISPs CHINANET and CNC-GROUP) comprise 63.9% of links to other countries. Furthermore, the same two ASes contain the great majority of filtering devices detected by the researchers, an entire 96.8%, while a small fraction were also found in provincial ASes. Overall, this shows that the bottleneck position of two ASes is used by the state for censoring purposes and highlights the importance of border ASes, while also raising questions about the role of internal filtering.

In a cross-national context, [26] maps the AS topology of different countries. The study finds that the structure of the topology can vary, with countries like China being more centralized, which is in line with the findings in [5], while other countries, such as Russia, are more decentralized. Nevertheless, for all nations, it was found that very few ASes have great control over the national network.

A further international comparison is conducted by [3], where the authors examine the relationship between a country's inter-domain routing topology and its freedom. Machine learning techniques are applied to predict the Freedom House Freedom of Press Index. The selected features include topological classic graph metrics, robustness, demographic metrics such as IP addresses per capita, and international connectivity. The study finds that they can predict the freedom of a country with high accuracy, with IP addresses per capita being the strongest indicator. AS topology features such as maximum path length to all other countries and high graph diameter were found to have a negative correlation with freedom of expression.

In [25], the authors measure the control key ASes exercise over the global Internet. They find that a mere 30 of approximately 50'000 global ASes intercept 90% of routing paths. Furthermore, it is discovered that one third of these ASes are located in nations known to censor Internet traffic, specifically China, Russia, and India. As many paths routed through these nations originate in other countries, the authors conclude that the impact of their censorship is not limited to their citizens and affects users internationally. On the other hand, in [27], the authors use this hierarchical organization to their advantage when placing decoy routers. Decoy routers help users evade censorship by covertly routing traffic to a secret destination. According to [27], due to the great impact of few ASes, only very few decoy routers need to be placed within non-censorious countries to great effect. Overall, these studies are aligned with aforementioned work, showing that control

is highly centralized within few ASes, while showing how this structure can be used to one's advantage in anti-censorship efforts.

Further examining censorship but with regard to a specific event, the authors of [28] investigate the HTTPS interception conducted in Kazakhstan in 2019 by experimentally triggering the interception mechanism. Among other things, they find that traffic had to pass through a specific part of the country's largest Internet Service Provider (ISP) in order to be intercepted. This shows that the effectiveness of this censoring method is dependent on and limited by topological control.

Again in a broader context, Lebya et al. [4] build on the research conducted in [5] by asking whether the borders of nations contain many access points or whether they act as chokepoints. Calculating the fraction of paths that route through a given border AS, the authors calculate the *National Chokepoint Potential (NCP)*. They find that the NCP varies highly across nations and find significant relationships between NCP and national press and Internet freedom. Furthermore, they identify a trend showing that AS topology has evolved over time to reflect national borders more closely. Relating this back to previously mentioned research on SCION, this may exacerbate the concern raised by [6] that SCION's ISD structure may afford a higher level of control to nation states.

Further building upon research on AS border topology, [2] focus their study on a specific event in Iran, when the government was able to selectively halt most international Internet traffic, while keeping the domestic network operational. The researchers conclude that the Iranian AS topology facilitates selective censorship, with most of the international traffic being connected through only three main ASes, all of which are controlled by the government. At the same time, the domestic network of Iran is highly decentralized and complex. The authors posit that this structure enables the Iranian government to exert control over international Internet traffic, while keeping domestic connections functional.

Similarly, [29] extends on the relation of AS topology and censorship. The authors introduce a metric called *funneling*, which measures the relative share of downstream ASes one AS serves. This value is averaged and applied recursively until no more downstream ASes are available. A high value for this metric implies that few ASes provide the connectivity for most domestic ASes. The study evaluates this metric across countries and finds a correlation between the funneling effect and Internet censorship.

While the previously discussed work has placed emphasis on centralized censorship, a further study shows that decentralized methods are also common [30]. The study finds that national censorship is often highly inconsistent. This suggests that path diversity is a valuable network feature that could allow users to circumvent censorship. Furthermore, it shows that fine-grained domestic Internet structure plays an important role in censorship.

Overall, the previous research supports this work's assumption that there is a correlation between Internet topology and censorship potential. Furthermore, it shows the importance of border ASes and their influence on their countries censorship potential. Other studies also show that censorship is often applied inconsistently. This work differs from the summarized work as it is mainly concerned with extending the topological analysis from the current state of the Internet to Future Internet Architectures. Focusing on the inherent

censorship potential of certain topologies, taking a predictive approach using *a priori* metrics is consistent with the goals of this study.

3.3 Rewiring for Robustness Optimization

This section reviews previous work on edge rewiring algorithms for maximizing network robustness, covering research from approximately the past 15 years.

The rewiring problem can be decomposed into two subproblems [31]: (1) Which non-existing edge should be added to maximally increase network robustness, and (2) which existing edge should be removed to minimally decrease it. These subproblems can be addressed in isolation or in combination. While many works address the subproblem of edge addition, this review is limited to algorithms that address both subproblems.

Finding the optimal set of edges to delete and add to enhance robustness is an NP-hard problem, as it subject to combinatorial explosion and greedy solutions are not guaranteed to find the global maximum. This also applies to related subproblems. For instance, finding a set of edges $A, e \in A, e \notin E$ to maximally increase the algebraic connectivity of a graph has been shown to be NP-hard [32]. Therefore, numerous efforts have been made to design approximation algorithms.

It is well understood that scale-free networks, such as the Internet, are robust against random failures but vulnerable to targeted attacks [33]. In their work [34], Schneider et al. conduct random, degree-preserving edge swaps on real-world scale-free networks. A random swap is applied only if it increases the robustness of the graph. The authors found that with a relatively small number of swaps, robustness can be increased significantly. Furthermore, the authors observe that the robust networks constructed by the algorithm show an "onion-like" structure. [35] expand on the random algorithm by favoring links between lower and higher degree nodes. The authors find that this mechanism similarly produces onion-like structures but also results in a faster increase in robustness. A further variation of the algorithm proposed by [34] is presented in [36], where swaps with a negative impact on robustness (or other cost functions) can still be accepted with a probability that is relative to the magnitude of the impact, as well as the number of edges that have already been swapped. This allows the algorithm to escape local maxima.

[37] propose four efficient capacity optimization algorithms: Disassortativity and Preferential Attachment (DPA), Disassortativity and Eigenvector Centrality (DEC), K-core (using Degree) and Betweenness Centrality (DKBC), K-core (using Closeness centrality), and Disassortativity and Betweenness Centrality (CKDBC). In DPA, edge (i, j) is removed if i and j are highly assortative. A new edge is formed between i and v, where higher degree nodes have a higher chance of being selected. DEC is a variation of this, where the newly established edge is selected based on eigenvector centrality rather than degree. DKBC splits nodes into k-cores based on degree. Furthermore, it evaluates the betweenness centrality of each node beforehand. A random edge (i, j) is deleted. A random node v is selected, and if v has a higher k-core and lower betweenness centrality than i, the edge (i, v) is added. CKDBC establish k-cores using closeness centrality. It then

selects a node i from the innermost core and its neighbor j, if i and j are assortative. The node v is then chosen if it is disassortative to i and if it has a smaller betweenness than i.

[38] introduce an edge-rewiring method while preserving the overall degree distribution of the graph. The authors simulate an attack by deleting random nodes. Thereby, edges are partitioned into valid edges (between two remaining nodes), invalid edges (between two deleted nodes), and flexible edges (between a remaining and a deleted node). Among other things, the rewiring algorithm replaces a valid edge (i, j) and a flexible edge (k, l) with edges (i, k) and (j, l), hence increasing the size of the remaining network. The authors find that this method outperforms previous strategies.

In [39], the work focuses on rewiring a graph that has been fractured into several components under edge attacks, by choosing two nodes from different components and deleting an edge within a different component. The authors tested four methods of selection, namely random rewiring, preferential attachment, clustering coefficient based rewiring, and k-core based rewiring. Within the context of their evaluation, namely dense social interaction networks, the preferential attachment method performed the best.

In contrast to the previously mentioned works, [40] designed a memetic algorithm that emulates genetic evolution. On the basis of an initial graph G_0 a population Ω is generated by randomly deleting and adding edges in G_0. These variations represent the *chromosomes*. So-called *child chromosomes* are formed by randomly selecting properties of two parents. Combined with a local optimization strategy, new graphs are thus generated and optimal topologies are selected based on their robustness values. The authors found that this algorithm outperformed previously proposed strategies.

Another heuristic is used in [41], where the authors leverage the relation between loops and robustness. The set of vertices which, if removed, render a graph acyclic is called the *Feedback Vertex Set* (FVS). Finding this set is NP-hard. However, the likelihood q_i^0 of a vertex i being in FVS can be approximated. In the proposed algorithm, edges between nodes with the highest q_i^0 are deemed most expendable and are replaced by new edges between nodes with low q_i^0. The authors implement both a degree-preserving and a non-preserving variation of this algorithm. They find that both perform at least as well as established alternatives. The non-preserving variation significantly outperforms standard, degree-preserving methods and results in graphs with much smaller differences in degree across nodes. This suggests that scale-free networks are overall unfavorable.

Unlike the previously discussed methods, spectral graph measures can also be leveraged by optimization algorithms [31], [42]. In [31], the authors introduce a heuristic to approximate the impact of each edge $e_{old} \in E$ and $e_{new} \notin E$ on algebraic connectivity if removed or added, respectively. The impact for a given edge $i(i, j)$ is estimated based on the lower and upper bounds of the algebraic connectivity, both of which contain the term $\alpha_{ij} = |\mathbf{u_i^{(2)}}(G) - \mathbf{u_j^{(2)}}(G)|$. Hence, $e_{\text{old}} = \arg\min_{i,j} \alpha_{ij}$ and $e_{\text{new}} = \arg\max_{i,j} \alpha_{ij}$. After each rewire, the algorithm recalculates the μ_2 and $\mathbf{u}^{(2)}$. The authors of [42] expand on this method in three ways: (1) definitions of α_{ij} are derived for various spectral robustness metrics, namely the spectral radius, the spectral gap, effective resistance, and number of spanning trees; (2) matrix perturbation theory is employed to update the eigenpairs of the adjacency matrix and the Laplacian, rather than recalculating from scratch; and (3)

the authors introduce an edge selection procedure that additionally preserves the degree distribution.

While various efforts have been made to optimize general networks, no research has been conducted on their applicability to FIAs. More specifically, the SCION Internet architecture introduces several connectivity constraints that an optimization algorithm must additionally enforce while maintaining performance.

Chapter 4

Theoretical Analysis

This chapter contains the theoretical evaluation of Internet topologies, whereby three different types of networks are examined. Graph theoretical concepts which are applicable to censorship resilience are gathered and presented in an extensive survey. In a second step, the selected metrics are applied to the following networks. The real-world AS topology based on CAIDA datasets is compared to a realistic SCION topology based on the same empirical data. Furthermore, both are compared to expander graphs, which act as an idealized model of connectivity and serve as a baseline.

The goal is to conduct an analysis that is largely independent of specific censorship strategies by using *a priori* metrics that indicate resilience regardless of both the number and location of censoring nodes within the network. The goal of this chapter, therefore, is to evaluate a network's *general censorship resilience potential* in a quantitative manner.

The chapter is organized as follows. First, the threat model is specified. This is followed by an extensive survey on graph theoretical methods. Subsequently, the methodology is introduced, detailing the data sources and processing steps. Finally, the results of the evaluation are presented and discussed.

4.1 Threat Model

Many censorship and surveillance techniques fundamentally rely on the ability to intercept traffic. For example, packet inspection, packet manipulation, and traffic analysis require traffic to traverse an adversarial network node or controlled link. This analysis assumes a censoring entity with the following capabilities: The censoring entity partially controls the network infrastructure. If compromised infrastructure is traversed, the traffic is exposed to a censoring apparatus.

In contrast to Wrana et al. [6], this work does not investigate techniques for censorship and surveillance once traffic is intercepted, but employs a graph-theoretical approach and focuses on a network's inherent qualities in resisting censorship. Furthermore, compared to Ivanovic et al. [7], this work is more general with regards to the presumed threat and censoring entities within the network.

4.2 Graph Robustness Metrics

The following survey on graph theoretical methods is conducted as a consequence of the given threat model definition:

The threat model implies that resilience to censorship is equivalent to robustness to failure in network science.

This equivalence arises from the property that a node's influence over a network describes both its effectiveness as a censoring node as well as the amount of disruption caused in the event of its failure. Similarly, a edge's censorship potential is equivalent to its impact in case of failure. This opens up a considerable corpus of research, as robustness to failure in network science is a well-studied concept.

Graph theory provides a wide range of metrics to evaluate the robustness of a network [43], [44]. Metrics measuring robustness differ in various dimensions, such as graph type, scope, robustness definition, and methodology (including analytical, procedural, and non-deterministic methods).

While excellent surveys on robustness metrics precede this work [43], [44], this work makes the following contributions: it narrows down the definition of robustness and is simultaneously more extensive, as the research combines sources from different fields, such as theoretical math and empirical censorship research. Furthermore, it introduces a new, more holistic categorization, e.g., by extending taxonomies inspired in part by [43] through the distinction between *a priori* and *a posteriori* methods. Finally, this summary helps disambiguate the literature by identifying duplicate and strongly overlapping metrics.

In this section, the robustness metrics are collected, compared, and categorized. The summary is preceded by a discussion of the criteria considered when selecting relevant graph metrics.

4.2.1 Criteria

Given the large number and diversity of graph metrics, they are narrowed down using a specific set of criteria. These criteria concern the types of graphs considered, the scope of the metrics, and the aspects of robustness they measure.

Layer Model

The Internet layer model is a well-established model of the structure and functionality of the Internet. There exist two main variants: the Open Systems Interconnection (OSI) model and the TCP/IP model. This work is concerned with the network layer and the Internet layer. For Internet-specific metrics, this work is limited to single-layer models that abstract away mechanisms beyond routing.

Graph Type

When modeling Internet topology, a variety of strategies exist, each serving a different purpose. For example, one can model connection bandwidth using edge-weights. It is also common to model customer-provider relationships using directed edges. This work aims to be broadly applicable to censorship-resilience properties in the Internet topology; therefore, only metrics defined on simple graphs are considered in the summary.

Scope

In graph theory, local metrics characterize specific vertices or edges within a graph, while global metrics describe the network in its entirety. This also includes global aggregates of local metrics. The survey of existing metrics is limited to global scope metrics, as the research is primarily concerned with examining a topology's resilience with regards to censorship. Local metrics can help to identify an architecture's vulnerabilities, but are not suitable to evaluate the overall architecture.

Robustness Aspect

Robustness encompasses various properties. The authors of [43] identify four robustness aspects: *Disconnection, Transmission Speed, Traffic*, and *Backbone*.

Disconnection robustness is concerned with whether any given pair of vertices can communicate (represented by a binary rather than scalar value, e.g., with respect to bandwidth) as well as the reliability of this connection in terms of path redundancy. Transmission Speed robustness describes a network's ability to keep paths short overall, while Traffic robustness relates to a network's ability to transmit large traffic volumes. Both aspects are related to network performance (either speed or bandwidth) rather than the binary ability to establish connection. The Backbone aspect is concerned with the Internet Backbone, a subset of nodes that form the core of the Internet. This aspect is too specific and depends on a secondary Backbone definition. The definition of robustness used in this survey is limited to what [43] define as Disconnection robustness.

The established definition disregards the dimensions of distance and speed. This is notable, as certain censorship methods do involve speed throttling. Nevertheless, for this work the conscious choice was made to abstract these dimensions away in the pursuit of simplicity. The metrics presented here are limited to those applicable to Disconnection robustness.

A notable consequence of this criterion is that it eliminates metrics that involve betweenness centrality—a well-established metric for measuring bottleneck structures—as it considers only shortest paths. In this context, where Disconnection robustness and path redundancy are central, betweenness centrality would introduce unwanted ambiguities. Furthermore, metrics like diameter and average shortest path length measure how spread-out a graph is. These metrics can be used in the context of connectivity (e.g. [3]).

However, it is the position of this work that the diameter is simply correlated with the connectivity. As an example, a graph of constant $|V|$ and $|E|$ has a more highly connected topology the lower its diameter, as its structure becomes less and less "chain-like". Therefore, these metrics are considered *indirect* metrics for connectivity and are omitted in favor of more immediate metrics.

4.2.2 Categorization Approach

The following section presents categories of metrics that were identified over the course of the research. Figure 4.1 visualizes the taxonomy and a full overview of all metrics can be found in Table 4.1.

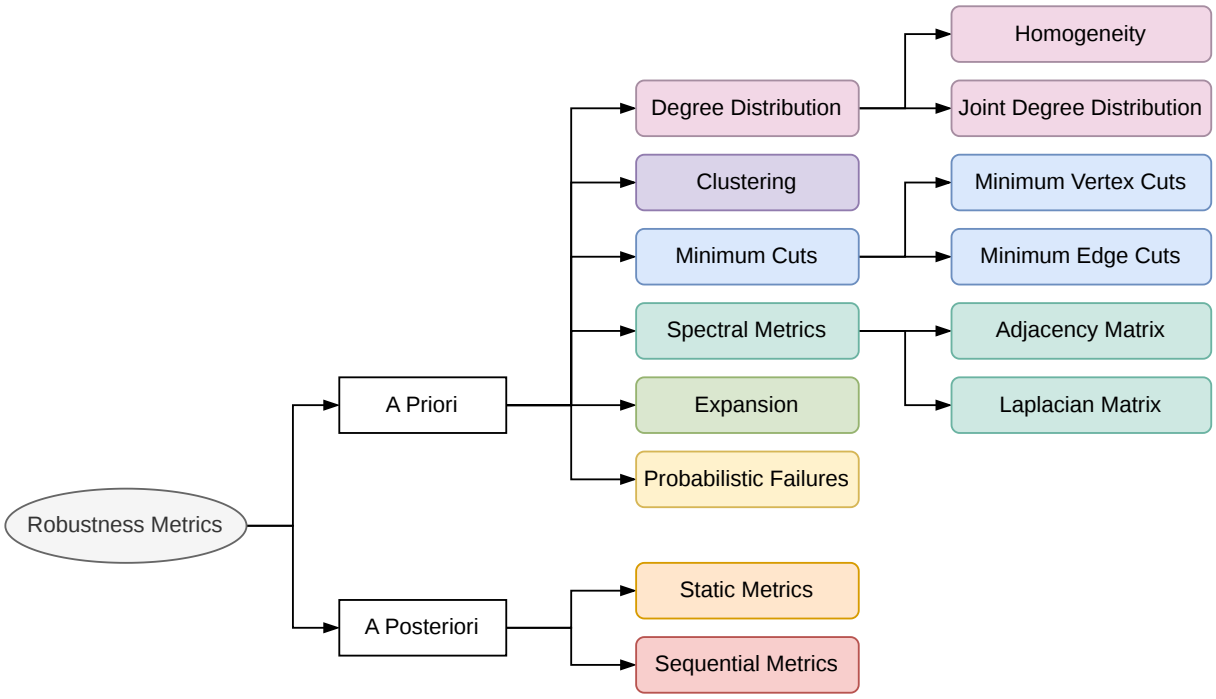

Figure 4.1: Visualization of the proposed taxonomy of robustness metrics.

Overlapping Metrics and Duplicates

A further goal of this summary is to identify *duplicate metrics* or *overlapping metrics* in order to disambiguate the literature, that is, to identify metrics that are equivalent or closely related in a mathematical sense. Overlapping metrics are grouped together in the table, and one representative is chosen. The related metrics are displayed indented underneath the representative one. In the taxonomy, grouped metrics and subcategories

differ semantically. In the table, they are visually distinguished, with group members not being numbered. Some of the groupings constitute a close relation, while others represent a one-to-one equivalence, and for some it depends on the application. All of these groupings signify, however, that the comprised metrics are *not meaningfully distinct*.

A Priori vs A Posteriori

Lou et al. [45] present a survey of *a posteriori* metrics and define *a posteriori* metrics as the practice of recording certain characteristics of the network over the course of an attack simulation. While these constitute a significant part of *a posteriori* methods, this work widens the definition to include methods that evaluate the impact of a fixed set of compromised or failing entities. In the context of robustness evaluation, *a priori* methods constitute a predictive evaluation of a network's vulnerability in the case of hypothetical failures. The latter are descriptive methods, characterizing the effect of failures after they have occurred. Translating this to the context of censorship, *a posteriori* methods require the censoring nodes or compromised links *to be known*, while *a priori* methods are fully agnostic the roles of specific nodes or links within the network.

Formally, *a posteriori* metrics include all metrics that evaluate robustness given a compromised set of nodes $\mathcal{X}$. The subcategories of *a posteriori* metrics are split according whether or not $\mathcal{X}$ is static. Consequently, in *Static Metrics*, the set $\mathcal{X}$ is fixed and known beforehand. In *Sequential Metrics*, the evaluation is conducted by taking measurements as nodes are sequentially added to $\mathcal{X}$, for example during an attack simulation. The criteria for a node's inclusion in $\mathcal{X}$ differ across metrics and applications. In some cases, nodes in $\mathcal{X}$ are randomly selected, while in others it is a domain-specific predetermined set, such as nodes representing border ASes. This categorization is designed to be agnostic to how the compromised set is derived and relies on the mathematical abstraction $\mathcal{X}$.

4.2.3 List of Categories

This section lists all categories, starting with *a priori* metrics, before transitioning to *a posteriori* methods. Category headers are annotated accordingly.

Degree Distribution (*a priori*)

Several works have proposed to evaluate the link between the degree distribution $P(k)$ (which measures the occurrence of a given degree k) and graph robustness [46]. Many empirical studies have suggested that the Internet topology follows a power-law distribution [47], characterized by major hubs and a heavy tail. Nodes therefore have a somewhat heterogeneous degree distribution. Wang et al. [46] suggest a link between network vulnerability and heterogeneous degree distribution and measure the former through *Degree Entropy*. Similar approaches are employed by [48], who introduce *Skewness*, and [49], who introduce the *Vulnerability Function*. Similarly, [50] use *Degree Variance* and [51], [52] use *Degree Balanced Graph Metric*. All of these metrics essentially attempt to measure

a graphs homogeneity (or, heterogeneity) with regard to degree distribution, and link it to a graph's robustness. Thus these metrics can be summarized by the subcategory of *Homogeneity*.

A further, related subcategory is the *Joint Degree Distribution*. In contrast to the degree distribution $P(k)$, the joint degree distribution $P(k|k')$ measures the probability that a node of degree k is connected to a node of degree k'. A well-established metric of joint degree distribution is the *Assortativity* of a graph. It measures whether nodes are more likely to be connected to nodes with similar degrees (assortative) or dissimilar degrees (disassortative). Past evidence suggests that the Internet is disassortative [53]. Furthermore, [54] study the assortativity of networks and find that assortative networks are more robust. A very similar metric is the *Average Neighbor Connectivity* introduced by [55].

Clustering (*a priori*)

Local clustering is another well-known metric from graph theory. Given a node i, it describes how tightly connected i's neighbors are among themselves. Within the global scope, this metric can simply be averaged across all nodes, as is the case for *Global Clustering* $\langle C \rangle$, or averaged across nodes with a degree > 2, as is the case for *Global Clustering* $\bar{C}$. *Transitivity* $C\Delta$ is an exclusively global metric that measures the ratio of total number of triangles to triplets (node sets of three, directly connected by at least two edges) in the graph [53], [56]. Since clustering is statistically related to degree correlations (as captured by the joint degree distribution), [57] introduce two clustering metrics that are independent of degree correlations. These are listed in Table 4.1 as *Correlation-Independent Transitivity* $\tilde{C}\Delta$ and *Correlation-Independent Clustering* $\tilde{C}$.

Global clustering is also highly related to the notion of *Modularity*, which measures the extent of a network's community structure, i.e. the presence of densely connected subgroups that are sparsely interconnected. Given a graph G of fixed $|V|$ and $|E|$, the graph starts exhibiting a stronger community structure as global clustering increases. *Modularity* [58] is another method of measuring community structure. A further method that identifies the extent of communal structure is *Spectral Cluster Identification* [59]. This robustness taxonomy includes a category for spectral metrics. However, as the spectral method here is applied as an intermediate step to identify community structure, it was deemed more applicable to this category.

Counterintuitively, high global clustering is a negative indicator of robustness when compared to other networks of the same size and edge count. High global clustering given a limited number of edges indicates highly separated communities. This in turn implies the existence of bridge edges [43]. This raises the question of interdependence across metrics. It is possible that, given this relation, the correlation of clustering and robustness is better measured by path redundancy and minimum edge cuts.

Minimum Cuts (*a priori*)

The *Minimum Cut Problem* is a fundamental problem in graph theory. A minimum cut is the smallest set of nodes (or edges) that must be removed in order to disconnect a graph. The *Max-Flow Min-Cut Theorem* states that the maximum flow (the maximum capacity that can flow from any given source node s to any sink node t) of a network is equivalent to the summed weight of its minimum edge cut. This implies that minimum cuts can be viewed as the dual notion of path redundancy [60]. Menger's theorem states that "the minimum number of points separating two nonadjacent points s and t is the maximum number of disjoint s-t paths" [61, Th. 5.9]. A common modern rephrasing states that "in a k-connected graph, every pair of vertices are joined by k internally disjoint paths" [62]. The property *k-connected* here refers to k-vertex-connectivity, meaning at least k vertices must be removed before a graph can be disconnected. This shows that the minimum cut problem is dual to path diversity [63]. Therefore, this category includes minimum cut and path redundancy metrics.

Vertex Connectivity and *Edge Connectivity* are fundamental metrics but often yield trivial results in practice. For example, many large real-world networks may contain many single-degree nodes and thus exhibit an edge connectivity of 1, making it less useful for comparing topologies. *Conditional Connectivity* is a type of *Vertex Connectivity* [64] (or *Edge Connectivity* [64]) extends the previous metrics defining any condition P for the resulting subsets [65].

Another way to avoid trivial cuts is used by *Sparsity* [66], which involves the ratio of the number of cut vertices to the size of the smaller resulting component. The *Sparsity* is defined as the global minimum of this ratio. The *Cheeger Constant* [67] is the analogous edge cut metric and is defined as the minimum ratio of the number of cut-edges to the size of the smaller subgraph across every possible partition in a graph. The *Isoperimetric Number* and *Edge Expansion* are well-established synonyms of the Cheeger constant. A formal definition of the Cheeger constant is given in Section 4.3.4. Both Sparsity and the Cheeger constant are global minimum metrics across all subsets of a graph, meaning that a straight-forward evaluation is NP-hard. However, [43] state that the Cheeger constant can be approximated by the Fiduccia-Mattheyses algorithm [68].

Further, [69] discuss *Tenacity* and introduce *Edge Tenacity*. *Tenacity* is the minimum of the size of an edge cut A plus the size of the largest connected component which remains. This is then inversely weighted by the number of connected components in $G - A$. *Edge Tenacity* is the edge-cut equivalent. [70] expand on k-connectivity by measuring the percentage of k-connected nodes for, and sum over all possible k. They introduce their *Resilience Factor*, also referred to as *Partial k-Connectivity*. The *Ratio of Disruption* measures the size of one component divided by the size of the edge cut and the size of the other component [71].

In [62], the authors propose to measure the *Average Connectivity* by measuring the maximum k-connectivity between all pairs (u, v) and normalizing it. Similarly, the *Average Network Flow* [72] (if unweighted, as is assumed here) would be the equivalent *Average Edge Connectivity*.

The *Number of Spanning Trees* measures the path redundancy of a network [73]. It can notably be derived by spectral methods, but is placed in this category due to its semantic meaning. The *Treewidth*, used in this context by [74], measures how close a graph is to a tree. As a graph approaches a tree structure, it contains fewer cycles and more bridge edges, whose removal disconnects the graph. This increases vulnerability, as it implies lower redundancy and smaller edge cuts. [75] introduce *Total Graph Diversity*, which is an aggregated path diversity metric that measures both the number of paths and their level of disjointness between two vertices.

Spectral Metrics (*a priori*)

Spectral Metrics are a well-known category of metrics that describe a graph's spectrum, based on either the graph's adjacency matrix A or its Laplacian matrix L. We denote the eigenvectors of A as $\{\mathbf{u}_i\}_{i\in[n]}$ and the eigenvalues as $\{\lambda_i\}_{i\in[n]}$, where $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ (sorted in descending order). Similarly, we denote the eigenvectors of L as $\{\mathbf{v}_i\}_{i\in[n]}$ and its eigenvalues as $\{\mu_i\}_{i\in[n]}$, where $0 = \mu_1 \leq \mu_2 \leq \cdots \leq \mu_n$ (sorted in ascending order). The *Spectral Radius* is equal to λ_1. A large λ_1 indicates a well connected graph [76]. The *Spectral Gap* is equal to $\lambda_1 - \lambda_2$ and is often used in used in robustness analysis, where a larger spectral gap implies better robustness [77]. The *Natural Connectivity* is the natural logarithm of the sum of all $e^{\lambda i}$, given by $ln(\frac{1}{n}\sum_{i=1}^{n} e^{\lambda_i})$ and can be considered the average eigenvector [42]. The *Algebraic Connectivity*, often $a(G)$, is given by μ_2. A larger μ_2 indicates a more robust graph [42]. However, it has the downside that it does not strictly increase as edges are added [56]. Furthermore, it is strictly 0 for disconnected graphs. The *Effective Graph Resistance* is given by $\sum_{i=2}^{N} \frac{1}{\mu_i}$ and the sum of the reciprocals of all eigenvalues. A smaller value therefore indicates better connectivity.

Expansion (*a priori*)

The local *Expansion* of a node i, parameterized by an integer h, is given by the fraction of nodes which are reachable within h hops. The metrics in this category are global aggregations of this local notion. The relation to connectivity is intuitive: the larger a node's h-hop environment, the better the local connectivity, with the same being true on a global scale. The metric simply called *Expansion* is a global average of local expansion [43]. This metric is related, but not equivalent to *Edge Expansion*, a synonym for the *Cheeger Constant*. *Effective Eccentricity* is another local metric with a global variant, which measures the fraction of nodes reachable from starting node i, within a given distance [78]. The average global aggregate closely corresponds to *Expansion*. Similarly, the *Effective Diameter* measures how many nodes are connected through a path of a given distance [78].

Good Expansion is a binary property. If a graph has good expansion, any subset of nodes S has a neighborhood larger than $\alpha \cdot |S|$ [43], [77]. [77] employ spectral methods to determine whether or not a graph satisfies the *Good Expansion* property, called *Spectral Scaling*. This method uses all eigenpairs of the adjacency matrix A, leading to high

complexity. The *Generalized Robustness Index* is an approximation of *Spectral Scaling* that uses only the k largest eigenvalues and their corresponding eigenvectors [79].

In [3], the authors use tree balance to predict the vulnerability of a graph, which measures the balance of the tree rooted at the highest centrality node. This metric is listed under *Expansion*. The reasoning for this goes as follows: the imbalance of the tree is essentially a measurement of a tree's depth, given a fixed number of nodes. The more shallow the tree, the greater is the expansion of the highest centrality node.

Probabilistic Failures (*a priori*)

Metrics in the *Probabilistic Failures* category evaluate graph vulnerability under uniform edge failure probabilities. They are therefore not considered *a posteriori* methods, as the failures are hypothetical and thus the result can be calculated analytically without assuming a set of failed nodes.

The *Reliability Polynomial* was introduced by [80]. It is given by $Rel(G) = \sum_i^{|E|} a_j p^j (1 - p)e - j$, where p denotes the probability of an edge functioning reliably [43]. Further, [81] introduce a general *Survivability Function*, which similarly constitutes a sum of edge failure probabilities. Let $P[S = s]$ denote the probability of a given property S assuming value s under edge failures. From this, the authors [81] also derived the *Expected Survivability*, given by $\sum sP[S = s]$. The feature S can be freely defined. An example given by the authors is the fraction of nodes which remain connected to a given central node.

Static Metrics (*a posteriori*)

The first *a posteriori* category is comprised of *Static Metrics*. In contrast to the second category, *Sequential Metrics*, *Static Metrics* are computed at a single point in time for a given set of nodes $\mathcal{X}$.

The *Largest Connected Component (LCC)*—which reports the number of nodes in the largest component—is the most basic of such metrics. Although it could be considered an *a priori* metric, in practice it is typically used to measure the number of vertices in the largest component after the removal of a set of nodes $\mathcal{X}$ on an initially fully connected network.

Reachability is defined as the fraction of node pairs connected by a path over all possible node pairs in a graph [82]. Similarly to the LCC, this is more salient in an *a posteriori* context, as a fully connected graph always yields a *Reachability* of 1. This metric is often used in different variations and can be found in the literature under different names, such as *Flow Robustness* [43]. Another variant of this metric is used by Ivanovic et al. [7], who define the *Global Reachability Potential*; its context is more specific, as this metric evaluates how many node pairs in a national Internet topology can communicate without traversing a foreign AS node. In this case, the foreign AS nodes constitute the set $\mathcal{X}$. Another variant of *Reachability* is *Avoidability Potential* [7]. It can be viewed as a more restrictive notion of *Reachability*, measuring the fraction of node pairs (s, d)

that can communicate without passing through a predefined set of censoring nodes C. It additionally requires that $s \in S$ and $d \in D$, where S, D, and C are disjoint sets. As before, the predefined set C is analogous to the set $\mathcal{X}$.

Levett et al. [29] define the *Funnelling Effect*. In simplified terms, it measures the fraction of downstream inland ASes a given border AS connects to, and aggregates this over all border ASes. This metric is minimized if the connectivity is equally distributed across all border ASes, whereas a high metric indicates that fewer ASes connect a greater share of inland nodes, indicating funneling. Again, this metric relies on predefined sets of border, inland, and foreign nodes, where border nodes form set $\mathcal{X}$.

Leyba et al. [4] define the *National Chokepoint Potential (NCP)*. The *Chokepoint Potential* is a local metric used for border ASes of a country, and measures the fraction of in- and out-going paths that have to pass through this given node. The NCP is defined as the number of border ASes needed to control a given fraction of nodes f. This metric is considered an *a posteriori* metric due to analogous reasoning, i.e. it relies on a predefined set of border ASes, which form the set $\mathcal{X}$.

Finally, this work introduces a novel metric called *Border Breadth*, which is discussed in detail in Subsection 4.2.4. It is specific to the domain of SCION networks, as it is defined as the ratio of outgoing edges to the number of nodes in the ISD. With respect to categorization, *Border Breadth* can be classified as an *a posteriori* metric, since the core nodes in the SCION ISD can be conceptualized as the potentially compromised set $\mathcal{X}$ in this context. The metric is listed in the general table, as it could be applied in any scenario concerned with a subset of nodes S and bordering nodes $\mathcal{X} \subseteq S$.

Sequential Metrics (*a posteriori*)

The final category is comprised of metrics which record a given value over the course of the gradual decay of a network. The sequence of node or edge removals is often defined by uniformly random failures, but can also be given by targeted attacks. Here, *Sequential Metrics* are considered a subtype of *a posteriori* metrics, in contrast to Lou et al. [45], who define *a posteriori* metrics as *Sequential Metrics* only.

Schneider et al. introduce R [34], which is defined as $R = \frac{1}{N}\sum_{Q=1}^{N} s(Q)$, where $s(Q)$ is the fraction of nodes in the largest connected component, with Q being the number of nodes removed thus far. This metric is sometimes found under different names, such as *Remaining LCC* [45]. A very closely related metric is the LCC measured over the course of edge failures [34], [45].

Furthermore, [83] propose *Communication Robustness*, which is the fraction of node pairs that remain communicable over time. This metric is equivalent to *Reachability* being measured over the course of failures. This work therefore refers to this metric as *Sequential Reachability*, as shown in Table 4.1. Finally, [50] introduce the *Simplified Comparative Robustness*. It ignores non-dominant terms of the original metric, and is therefore not equivalent, but is shown to work almost the same in practice.

In their study, Singh et al. [3] use the *Integrated Algebraic Connectivity over Node Attacks*, which is "the area under the decay cruve [sic] of algebraic connectivity of the graph as nodes are removed in the decreasing order of their AS rank" [3]. The domain-specific property of AS rank can be generalized to any node ranking, making this metric applicable to other scenarios.

The *Critical Fraction* differs somewhat from the other metrics in this category, as it does not record an accumulated value over sequential removal, but rather a threshold value. It denotes the fraction of nodes that need to be removed so that the network decays, i.e. has no more giant component. Very similar to this is the *Percolation Threshold*, a metric from classical graph theory. In a network of nodes, the *Percolation Threshold* usually measures the fraction of edges which are *added* until a giant component *emerges*. Both metrics converge to a conceptually identical threshold, differing only in one being defined in terms of node removal and the other by edge addition. For both metrics, there exist explicit formulae, given that the graph fulfills certain conditions. In a more general case, though, the values are determined experimentally, making it an *a posteriori* method.

Metric	Citation
A Priori	
1 Degree Distribution	
1.1 Homogeneity	
1.1.1 Degree Variance	[72]
1.1.2 Entropy	[43], [46]
1.1.3 Skewness	[43], [48]
1.1.4 Vulnerability Function	[43], [49]
1.1.5 Degree-Balanced Graph Metric	[51], [52]
1.2 Joint Distribution	
1.2.1 Assortativity	[43], [54]
1.2.2 Average Neighbor Connectivity	[43], [55]
2 Clustering	
2.1 Transitivity $C\Delta$	[43], [44], [53], [56]
Global Clustering $\langle C \rangle$	[43], [84]
Global Clustering $\bar{C}$	[43], [57]
2.2 Correlation-Independent Transitivity $\tilde{C}\Delta$	[43], [57]
Correlation-Independent Clustering $\tilde{C}$	[43], [57]
2.3 Modularity	[43], [58]
2.4 Spectral Cluster Identification	[43], [59]
3 Minimum Cuts	
3.1 Minimum Vertex Cuts	
3.1.1 Vertex Connectivity	[43], [56], [60], [64], [72], [85]
3.1.2 Conditional Connectivity	[43], [65]
3.1.3 Sparsity	[43], [66]
3.1.4 Partial-k-Connectivity (Resilience Factor)	[43], [70]
3.1.5 Tenacity	[43], [69]
3.1.6 Average Connectivity	[62]
3.2 Minimum Edge Cuts	

Metric	Citation
3.2.1 Edge Connectivity	[43], [56], [60], [64], [85]
3.2.2 Conditional Connectivity	[43], [65]
3.2.3 Cheeger Constant	[43], [67], [86], [87]
Isoperimetric Number	[86], [88]
Edge Expansion	[88]
3.2.4 Minimum m-Degree	[43], [89]
3.2.5 Ratio of Disruption	[43], [71]
3.2.6 Edge Tenacity	[43], [69]
3.2.7 Number of Spanning Trees	[43], [44], [56], [72], [73]
3.2.8 Total Graph Diversity	[52], [75], [90]
3.2.9 Treewidth	[74]
3.2.10 Average Network Flow	[72]
4 Spectral Metrics	
4.1 Adjacency Matrix	
4.1.1 Spectral Radius	[44], [76]
4.1.2 Spectral Gap	[43], [44], [52], [77], [90]
4.1.3 Natural Connectivity	[43], [44], [72], [85], [90], [91]
4.1.4 Weighted Spectral Distribution	[52], [90], [92], [93]
4.2 Laplacian Matrix	
4.2.1 Algebraic Connectivity $a(G)$	[43], [44], [52], [56], [72], [85], [90], [94]
4.2.2 Effective Graph Resistance	[43], [52], [95]
Network Criticality	[43], [52], [90], [96]
5 Expansion	
5.1 Expansion	[43], [97]
Effective Eccentricity (global)	[78][43]
Effective Diameter	[78][43]
5.2 Good Expansion	[43], [98]
5.3 Spectral Scaling	[44], [77]
Generalized Robustness Index	[44], [79]
5.4 Tree Balance	[3]
6 Probabilistic Failures	
6.1 Reliability Polynomial	[43], [56], [80]
6.2 Survivability Function	[43], [81]
6.3 Expected Survivability	[43], [81]

A Posteriori

7 Static Metrics	
7.1 Largest Connected Component (LCC)	[43], [44]
7.2 Reachability	[43], [82]
Flow Robustness	[43], [52], [75], [90]
Avoidability Potential	[7]
Global Reachability Potential	[7]
7.3 Funnelling Effect	[29]
7.4 National Chokepoint Potential	[4]
7.5 Border Breadth	*This Work*

Metric		Citation
8	**Sequential Metrics**	
8.1	R	[34]
	LCC (per node attacks)	[34], [45], [72], [85]
	LCC (per edge attacks)	[34], [45], [85], [99]
8.2	Communication Robustness	[45], [50], [72], [83]
	Sequential Reachability	*Alias Introduced By This Work*
	Simplified Communication Robustness	[45], [50], [72], [83]
8.3	Integrated $a(G)$ over Node Attacks	[3]
8.4	Critical Fraction	[72], [100]
	Percolation Threshold	[33], [43]

Table 4.1: Overview of collected robustness metrics.

4.2.4 Border Breadth

In this part, the *Border Breadth*, a metric specific to SCION ISDs, is introduced. The Border Breadth measures the ratio of outgoing edges of an ISD over the nodes in the ISD. Given a graph G and a subgraph ISD corresponding to a specific SCION ISD, its Border Breadth BB_{ISD} is defined as follows.

$$BB_{ISD} = \frac{\partial(ISD)}{|V(ISD)|}$$

This metric quantifies how strongly the core ASes control connectivity between an ISD and the rest of the network, relative to the size of the ISD. A smaller value indicates stronger control, as it corresponds to fewer edges connecting the ISD to the outside relative to the number of nodes it contains. For a fixed number of boundary edges, larger ISDs imply stronger control, since more nodes are affected.

An advantage of this metric is that it measures the ratio of cut edges to the size of a partition, making it directly comparable to the Cheeger constant. The Cheeger constant is defined as the minimum such ratio over all possible partitions of the graph and captures the graph's weakest connectivity bottleneck. A formal definition of the Cheeger constant is given in Subsection 4.3.4.

This comparison thus assesses the role of the core ASes:

- If the Border Breadth of an ISD is smaller than the minimum Cheeger constant within the ISD, the core ASes present the tightest bottleneck in the ISD.
- If it is larger in comparison, this implies that more significant bottlenecks are present within the ISD itself.

4.3 Methodology

This section details the theoretical methodology of this work. It begins with a description of the datasets employed, followed by a comparison of the current AS topology with the SCION architecture. Specifically, it explains how the topological representations were derived in both cases. Finally, the results of the calculations are presented and discussed. The source code used for the theoretical evaluation is publicly available on GitHub [101].

4.3.1 Datasets

This work uses CAIDA data on the global AS topology as a basis for its evaluation. Specifically, the following datasets are used:

(1) *AS Relations*, Serial 2, December 1, 2025 [102]. This dataset contains a list of customer-provider and peer-to-peer relationships across ASes.

(2) *AS to Organization*, December 1, 2025 [103]. This dataset consists of two files: one mapping AS identifiers to organization identifiers among other metadata, and a second mapping organizations to further information such as organization name and country. The data is based on WHOIS records maintained by Regional Internet Registries (e.g., ARIN, RIPE NCC, APNIC) and National Internet Registries.

Following the method proposed by CAIDA [104] and adapted by Ivanovic et al. [7], a third dataset is constructed as follows:

(3) *AS Customer Cone Size*, December 1, 2025. The customer cone of an AS is the set of ASes, IP prefixes, and IP addresses which are reachable from the AS using only provider-to-customer links (i.e. by going downstream). The customer cone in this construction is simplified, using only the AS Relations dataset from December 2025. Thereby, the customer cone size here is limited to AS level topology only. The result is a dataset which maps AS identifiers to its total number of downstream ASes.

4.3.2 Base AS Topology

The base AS topology represents the current state of the Internet. It is mostly navigated by BGP routing—the basic, single-path inter-domain routing protocol that is mostly used today. While BGP routing allows for multiple paths to be learned (as the *AS Relations* dataset shows, path diversity does exist in the current AS topology), BGP selects a single best path per IP prefix. Alternatives to BGP do exist that support multipath routing, with SCION being the most notable architecture in the context of this work. However, SCION is not the only way of achieving inter-domain multi-path routing; BGP extensions such as *BGP Add-Path* expose multiple inter-domain paths [105]. Furthermore, routing decisions in BGP are not made by the communication end-points, but rather by intermediaries. Given the conditional multi-path capabilities of BGP routing, the AS topology reflects

maximum path diversity potential rather than a the actual set of paths available to a sender.

The *AS Relation* dataset contains 78'771 nodes and 723'215 edges. Therefore, the average degree of an AS node in the graph is approximately 18.36. Many of the metrics in Table 4.1 are highly complex. In order to conduct experiments on a topology of this size, graph sampling methods were used. The following section discusses the graph sampling in more detail.

Graph Sampling

Graph sampling methods can be a useful tool when analyzing large graphs, as they improve processing efficiency by reducing the volume of data [106]. Different types of sampling methods exist, such as Breadth-First-Search-Based Sampling Methods (BFS-SM), Node-Edge Sampling Methods (NES-SM) and Random-Walk Sampling (RWS) [106]. Node-Edge Sampling methods are arguably the most simple as they involve selecting nodes or edges uniformly at random across the entire graph. The method used here is a simple *Contraction of random vertex/edge (CRVE)* [107], where node i is selected uniformly at random. Then, a random neighbor of i is selected, node j. Finally, the edge between i and j is contracted. This is repeated until the desired reduction is attained. This has three advantages: (1) CRVE is not localized (unlike BFS-SM and RWS) and is better suited to capturing global structures. (2) The contraction method preserves basic connectivity. Since this work focuses on robustness analysis, disconnecting a connected graph might have an exaggerated effect on robustness analysis. (3) Ease of implementation.

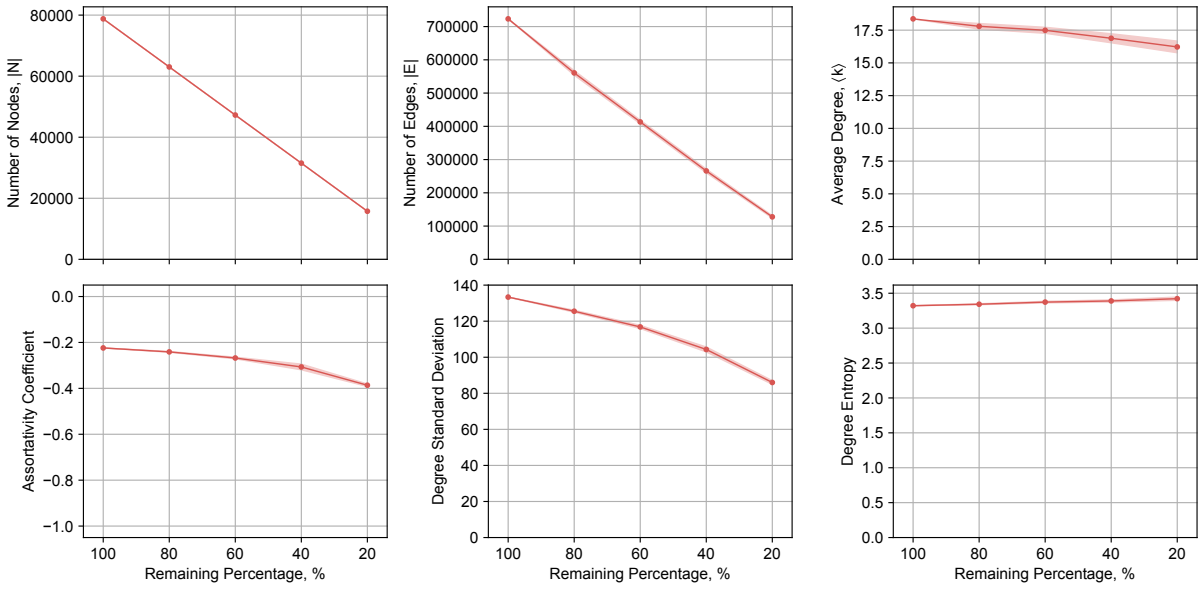

Figure 4.2: Basic graph metrics over the CRVE downsampling process, averaged across 10 samples.

Figure 4.2 shows basic graph metrics over the course of four reduction steps, corresponding to 80%, 60%, 40% and 20% of nodes remaining. For each reduction, 10 samples

were produced. The plots show the mean across the 10 samples, along with the standard deviation. Overall, there is a fairly low variance across the samples. Furthermore, this simple sampling method performs fairly well in terms of preserving the average degree of the graph, though there is a slow downward trend. The degree standard deviation decreased, while the degree entropy increased. This indicates that degree distribution becomes more homogeneous through CRVE. Furthermore, the original graph was slightly disassortative, meaning that high-degree nodes were somewhat more likely to connect to low-degree nodes. During downsampling, the graph became more disassortative. Overall, the average degree is expected to have the highest impact on robustness. As CRVE performs adequately in this regard, this method was deemed appropriate. In the forthcoming analysis, the current Internet topology is represented by a set of 10 CRVE samples of *AS Relations*, reduced to 20% of the original size.

4.3.3 SCION Representation

The SCION architecture is an FIA specific to the inter-domain routing. Among its most important features are the introduction of Isolation Domains and the built-in multi-path capability. Functioning real-world instances of SCION exist, such as the SCIERA network. However, the SCIERA network currently contains 29 nodes and is not comparable to the BGP network. Therefore, following the work done by Ivanovic et al. [7], the *AS Relations* dataset is used and ISDs are constructed on top in order to simulate a potential global SCION architecture. The construction procedure is detailed in the following steps:

(1) **ISDs.** Using the *AS to Organization* dataset, each AS's country is derived. Countries are used as a proxy to construct isolation domains. Given the intended purpose of ISDs as isolated trust-domains, oftentimes under a common jurisdiction [19], this is a reasonable heuristic.

(2) **Core ASes.** Each ISD in SCION has a set of core ASes that fulfill additional responsibilities, such as administering the ISD's policy, called the Trust Root Configuration. Furthermore, the core ASes are densely connected with core ASes of other ISDs, serving as natural bridge nodes. In line with Ivanovic et al. [7], the *AS Customer Cone Size* is used to determine which nodes are likely to take on the role of core AS.

(3) **Links.** As the topology is divided into ISDs and core ASes are designated, cross-ISD links connecting non-core ASes are disregarded, while intra-ISD links and links among core ASes are kept.

4.3.4 Expander Graphs and Network Partitioning

This theoretical evaluation aims to compare the resilience potential of the SCION topology to that of the current Internet. Both architectures are compared to *Expander Graphs*, which serve as the optimal benchmark in terms of resilience. Expander graphs are simultaneously sparse and highly connected, making them ideal communication networks [88]. Expander graphs are characterized by the Cheeger constant (see Table 4.1), also known

as the *Edge Expansion* or *Isoperimetric Number*. The Cheeger constant is therefore considered a vital metric in this work.

Let G be a graph. The Cheeger constant $h(G)$ of graph G is formally defined as follows.

$$h(G) = \min_{\substack{S \subset V \\ 0 < |S| \leq |V|/2}} \frac{|\partial S|}{|S|}$$

where S is a subgraph of G and ∂S is the set of cut-edges between S and $G - S$. This metric is a minimum over all subsets of graph G. A naive search for this minimum therefore is inherently exponential in complexity, with $O(2^n)$, where $n = |V|$. Fortunately, the Cheeger constant can be obtained by an implementation of the network partitioning algorithm proposed by [68]. The partitioning algorithm takes a graph G and parameters r and m and produces a minimum partition of a graph. Given a resulting bi-partition with subgraphs A and B, and $f = \frac{|A|}{|B|}, |A| \leq |B|$, then f approximates r. Thus r configures the ratio of the subgraphs. The parameter m defines the allowable deviation of f from r.

4.4 Results and Discussion

In this section, the results obtained from the theoretical evaluation are presented and discussed. First, a comparison using a diverse set of robustness metrics compares the different Internet architectures. Next, SCION ISDs are examined more closely, using the proposed metric *Border Breadth*. Finally, the findings are summarized and the limitations of this evaluation are discussed.

4.4.1 Comparison of Internet Architectures

Firstly, the current AS topology, the SCION architecture as well as the expander graphs are quantitatively compared in terms of a set of selected metrics contained in Table 4.1. The current AS topology is represented by downsampled probes from the original CAIDA *AS Relation* dataset. The samples were reduced in size by 80% from the original, using the CRVE method. As this is a significant downsampling, 10 probes are taken and averaged in order to counteract sampling error. The results shown in the plot represent said average, with an error bar displaying the standard deviation across the probes. The current Internet topology is referred to as "BGP CRVE 20%" in the results. The SCION architecture is split into national ISDs, as well as the SCION Core. The SCION Core is composed of the highly interconnected core ASes from each ISD. This results in a comparison both across SCION ISDs and architectures. Alongside basic metrics, specifically the number of nodes, the number of edges, and the average degree, Figure 4.3 also shows the assortativity, the degree standard deviation, and the degree entropy. The expander graphs were generated synthetically using the NetworkX [108] function `networkx.random_regular_expander_graph(n=1965, d=16, max_tries=10 000 000)`. The algorithm generates candidate regular graphs and iteratively tests whether they satisfy

the expander construction criteria. This rejection-sampling procedure can become computationally expensive for large graphs, motivating the use of a high upper bound on the number of attempts. In order to counteract the sampling error resulting from random generation, 5 expander graphs are sampled and averaged, with an error bar displaying the standard deviation across the samples. As the expander graphs exhibit very low variation, 5 probes were deemed sufficient. In the analysis, they serve as a high-connectivity baseline topology for comparison with the other network structures.

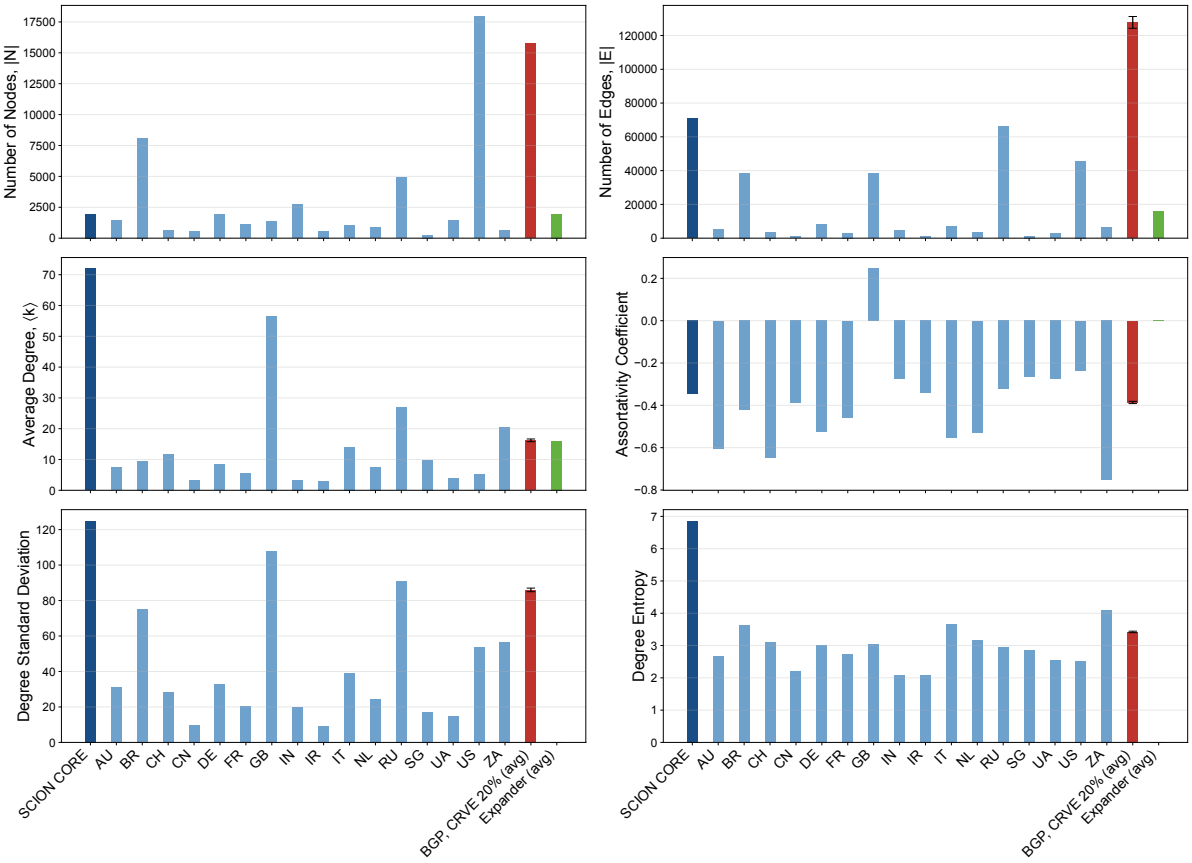

Figure 4.3: Simple graph metrics across SCION core (dark blue), SCION ISDs by country (light blue), the sampled AS topology (red) and expander graphs (green).

Figure 4.3 is consistent with many expected patterns. First, in line with past empirical data, the Internet is shown to be mostly disassortative, with the only assortative topology being the Great Britain ISD. Furthermore, the assortativity is undefined for the expander graphs, given that they exhibit a degree variation of 0. Other than that, the decomposition into ISDs has little effect on assortativity. The unusual value for Great Britain is likely linked to its abnormally high average degree. As Great Britain is a relatively small island country, it is expected to have few ASes. Its comparatively high number of edges could be explained by ISP policies or economic factors, such as a large IT sector compared to the country's area, as well as the island's geographic situation.

Furthermore, one can observe a correlation between average degree and the degree standard deviation. This is likely attributable to the fact that given a higher degree range,

the standard deviation is usually higher. While the degree entropy is conceptually similar to the standard deviation, it does not follow quite the same pattern. For example, Great Britain shows similar results to other ISDs and does not stand out. As expected, expander graphs have a degree standard deviation of 0 and an entropy of 0, as all nodes have the same degree.

Drawing the link to graph robustness, the entropy results are in line with expectations. The SCION core is highly connected and therefore most likely the most robust part of the topology. Accordingly, it has the highest degree entropy. In terms of degree standard deviation, the results contradict expectations, as the robust SCION core has a large standard deviation. However, this is explained by the correlation of the average degree and the variance. A comparison of standard deviation would be more salient if taken across topologies of equal average degree.

Figure 4.4 shows three more complex metrics: the Cheeger constant, the algebraic connectivity, and the spectral gap. The average degree is also included for convenient comparison.

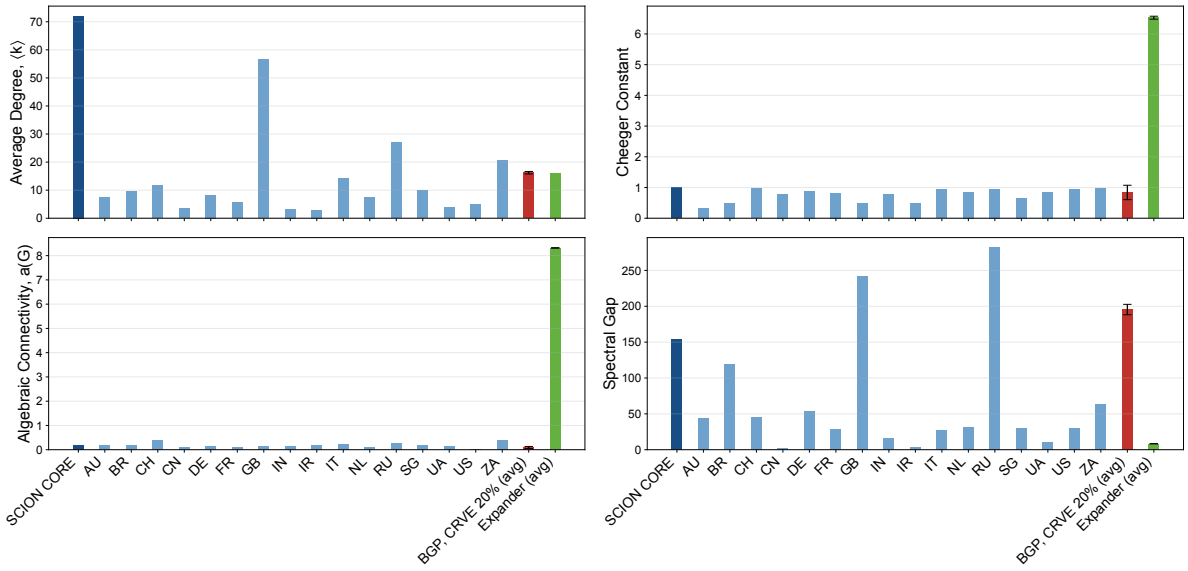

Figure 4.4: Complex graph metrics across SCION core (dark blue), SCION ISDs by country (light blue), the sampled AS topology (red) and expander graphs (green).

Firstly, the results show an extreme discrepancy in the Cheeger constant between the expander graphs and the other networks. This is expected to some extent; however, when accounting for the average degree, the difference becomes even more significant. Since the Cheeger constant measures the ratio of cut edges to the size of a partition, it is related to the average degree. Notably, the Cheeger constant is much higher for expander graphs than for the SCION core, despite the expanders having a much smaller average degree than the SCION core. This implies that it is possible to achieve much higher connectivity with the same number of edges across all non-expander topologies. In terms of the Cheeger constant, the SCION core and the ISDs perform similarly to CAIDA topologies overall.

Similarly, as shown in Figure 4.4, expander graphs exhibit dramatically higher algebraic connectivity than the other topologies. No clear pattern can be observed in the non-

expander topologies. Another very interesting observation is found in the spectral gap. The SCION core, Great Britain, and Russia have a large spectral gap. This is sensible, as these topologies also have a higher average degree and might exhibit greater robustness. However, the expander graphs are shown to have an extremely small spectral gap. Comparing these results with Figure 4.3, this may suggest that the spectral gap is sensitive to the graph size and edge count, which may limit its usefulness in scenarios involving graphs of varying scale.

Overall, the results are consistent with expectations, as they show that expander graphs are the most robust. The expanders exhibit a much higher Cheeger constant and greater algebraic connectivity. Across the remaining topologies, no meaningful pattern was identified with regard to these metrics. However, these results suggest that there is potential for substantial improvement in network connectivity without adding additional edges.

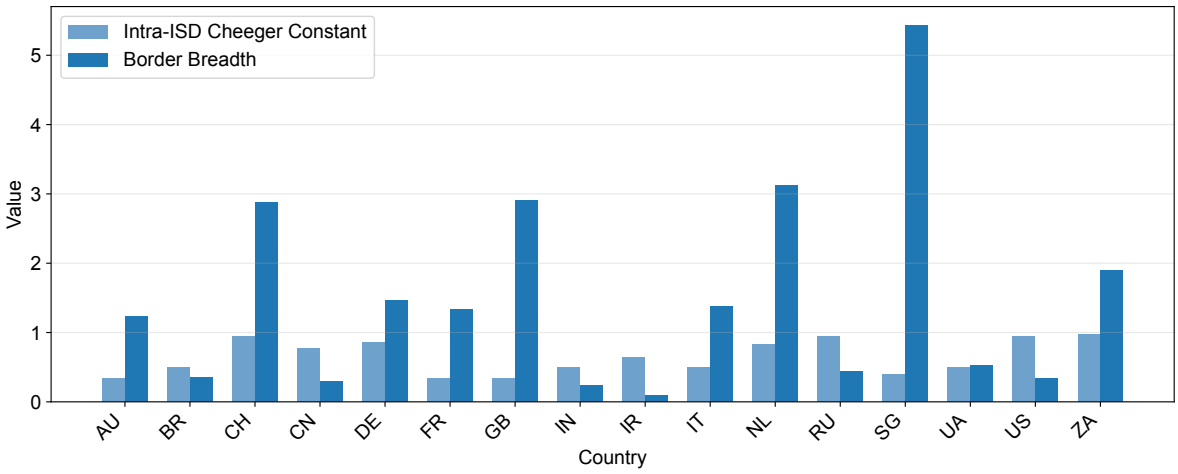

Figure 4.5: The Border Breadth compared to the intra-ISD Cheeger constant across country ISDs.

Finally, this analysis compares intra-ISD robustness to inter-ISD robustness using the Border Breadth. As outlined in Subsection 4.2.4, a Border Breadth that is smaller than the minimum Cheeger constant (within the ISD subgraph) implies that the border presents the strongest bottleneck structure for the ISD nodes. If a smaller Cheeger constant is present within the ISD, the border does not constitute the strongest bottleneck. Figure 4.5 shows a very high variation across countries with regards to Border Breadth. Singapore has a very large number of outgoing edges compared to its internal ISD size, which likely reflects geographic and economic factors. Interestingly, Russia, Iran, and China—countries which are known to exert censorious control over the Internet—have a Border Breadth metric smaller than their internal Cheeger constant, indicating that the core ASes would likely present the main bottleneck of their national architecture. A related analysis was conducted by [4], who found a correlation between Internet freedom and *National Chokepoint Potential*.

4.4.2 Summary and Limitations

Based on the theoretical evaluation, two key observations were made. First, a robustness metric comparison using expander graphs as benchmarks shows that many real-world topologies have clear potential for improvement without changing the number of nodes or edges. This suggests that strategic edge rewiring could significantly enhance censorship resilience. Second, Border Breadth varies substantially across countries and may be correlated with Internet censorship.

One important limitation to consider are possible sampling biases introduced by the CRVE downsampling of the *AS Relation* dataset. Another limitation of this analysis lies in the construction of the SCION topologies. Following the methodology employed by Ivanovic et al. [7], hypothetical ISDs were constructed based on countries, and ISD cores were constructed based on AS customer cone size. Although these are reasonable heuristics, they should be interpreted with caution, as they reflect a plausible, yet still hypothetical scenario. This limitation, however, also indicates an opportunity: as the global SCION topology is hypothetical, it can still be optimized for robustness. The next chapter therefore focuses on robustness optimization methods developed specifically for SCION topologies.

Chapter 5

Design

This chapter proposes a design for a censorship-resilient SCION architecture. Rather than growing a topology from scratch, the proposed solution can be applied to any existing SCION topology. Under strict constraints of keeping the nodes and number of edges constant, the goal is to perform the minimum amount of edge rewirings while achieving maximum resilience. The aim of these constraints is to generate feasible solutions in a context where SCION nodes are likely to be deployed on pre-existing physical networks where participants and their roles are likely predetermined. Furthermore, establishing new edges altogether in a communication network (and thereby increasing the edge count $|E|$), increases deployment and maintenance costs. Therefore, the aim is to maximize censorship resilience under the consideration of economic constraints.

In order to deploy a working SCION architecture capable of optimizing for robustness, the solution must fulfill the following requirements.

(1) The solution must contain a tool that finds optimal rewirings and outputs a modified topology.

(2) The solution must be able to automatically instantiate the topology generated in *(1)* as a functioning SCION network.

This chapter focuses on the algorithms employed to fulfill the first requirement. Steps taken to address requirement *(2)* are detailed in Chapter 6. The rest of this chapter is organized as follows. The first section details the network configuration constraints that SCION introduces. The second section discusses the reasoning for the metrics chosen as objective functions for the resilience optimizations. Finally, the last section introduces two algorithms designed to maximize their respective resilience metric, while enforcing SCION-specific constraints.

5.1 SCION Constraints

The SCION architecture is a Next-Generation Network that operates on the AS-level and constitutes an alternative to BGP. Its main purpose is to enhance global security by isolat-

ing trust among trusted ASes, so that routing information stays localized among trusted parties. A set of trusted ASes forms an ISD. Each ISD contains a subset of core nodes, which are responsible for administrative tasks as well as establishing governing policies. Further, core nodes are responsible for inter-ISD routing. The SCION architecture allows peer-links connecting non-core nodes across ISDs. Additionally, ASes can be members of several ISDs. However, this work excludes the last two properties, as they do not reflect the fundamental design choices of SCION. Moreover, this aligns with the structure of the real-world SCION network SCIERA [8]. Therefore, for this analysis, only core nodes can form connections to other ISDs, resulting in the following topological constraint:

C1 *Nodes $i, j \in G_{SCION}$ can connect if both i and j are core nodes or if both are located within the same ISD.*

Given a SCION network with n nodes, this constraint can be encoded within the connection matrix C. C is a binary $n \times n$ matrix, where each element C_{ij} indicates whether a connection between nodes i and j is allowed according to **C1**. Let $isd(i)$ map node i to its ISD, and $\mathcal{X}$ denote the set of all ISDs. $CORE_X$ constitutes the core nodes of an ISD X. Define $\text{CORE} = \bigcup_{X \in \mathcal{X}} \text{CORE}_\text{X}$ as the set of all core nodes across all ISDs. Then the matrix C is constructed as follows:

$$C_{ij} = \begin{cases} 1 & \text{if } i \neq j \text{ and } \big(\{i,j\} \subseteq \text{CORE or } \text{isd}(i) = \text{isd}(j)\big) \\ 0 & \text{otherwise} \end{cases}$$

$$\forall i, j \ \big((i,j) \in E(G_{\text{SCION}}) \implies C_{ij} = 1\big)$$

Note that in addition to the SCION-specific condition $\{i,j\} \subseteq \text{CORE or } \text{isd}(i) = \text{isd}(j)$, the formulation also prohibits self-loops. This is not specific to SCION, but applicable to the traditional Internet architecture.

C2 *The global combination of core nodes must form a connected subgraph.*

This constraint arises from the *valley-free* routing limitation [109]. This limitation states that a node should not provide transmission between two of its providers or two neighbors which share a peering connection. Essentially, a package may only travel upstream, downstream, or laterally through the network and may not traverse valleys by first traveling downstream and then upstream. As core nodes always form a provider-customer relationship with non-core nodes, interposing non-core nodes between cores disrupts their communication, as messages are forcibly routed through a valley.

In order to formally capture this constraint, let G_{CORE} be the induced subgraph of G_{SCION} induced by the vertex set $CORE$:

$$G_{\text{CORE}} \ = \ G_{\text{SCION}}[\text{CORE}]$$

Further, let $k(G)$ be the number of connected components in a graph G. Then, **C2** is satisfied if and only if:

$$k(G_{CORE}) = 1$$

Additional Constraints In addition to the SCION-specific constraints, more general topological limitations apply. As previously mentioned, reflexive links are excluded. Further, only connected graphs are considered viable. Most robustness metrics implicitly enforce this; however, it is still notable to point out in the context of censorship. As the goal is to facilitate information flow, a disconnected graph is considered invalid.

The next section further discusses the precise robustness metrics that were chosen in order to maximize resilience.

5.2 Selection of Objective Metrics

As shown in Chapter 4, numerous robustness metrics exist. Two metrics are selected as objective functions, each leading to the implementation of a distinct optimization algorithm. This section presents the rationale behind their selection.

Sequential attacks, being non-analytical procedures, are excluded as objective functions. Further, metrics which measure degree distribution and homogeneity are not used. While they correlate to robustness [34], [35], more precise and direct metrics exist. Spectral metrics are also good candidates for efficient and reliable robustness metrics. However, the metrics such as *Effective Resistance* or *Number of Spanning Trees* require the full set of non-zero eigenvalues of the Laplacian matrix. A more promising alternative is *Algebraic Connectivity*. It is often used in robustness analyses and is numerically easier to calculate. Furthermore, the analysis in Chapter 4 showed much potential to increase algebraic connectivity without increasing graph density.

Algebraic Connectivity The algebraic connectivity of a graph G, $a(G)$, is defined as the second smallest eigenvalue μ_2 of the Laplacian matrix L. It reflects the extent to which a graph can be decomposed into weakly connected components. Small values of μ_2 imply sparse cuts, and hence the presence of severe bottlenecks. Among other properties, $a(G)$ is 0 if a graph is disconnected. This property can be used to enforce the connectivity constraint among candidate solutions.

Therefore, algebraic connectivity is a useful metric due to its clear relation to robustness and its ease of implementation and computation.

Cheeger Constant The Cheeger constant is a normalized minimum cut [67]. It is defined as the minimum ratio of cut edges over the smaller cut size resulting from a graph partition:

$$\min_{\substack{S \subseteq V(G) \\ 0 < |S| \leq \frac{|V(G)|}{2}}} \frac{|\partial(S)|}{|S|}$$

where $\partial(S)$ denotes the set of edges crossing the cut. A small Cheeger constant implies the presence of a severe bottleneck, whereby the number of edges forming the bottleneck is weighed against the portion of the network they control. Considering this ratio is useful, as unweighted minimum cuts are often trivial for real-world networks. Consider, for example, that a graph with a single 1-degree node always has a minimum edge cut of 1. A minimum edge cut where 3 edges sever 2 nodes from a graph results in a larger Cheeger constant than a cut of 3 edges severing 50 nodes, which has an extremely small Cheeger constant and much more severe bottleneck.

Many metrics in the domain of minimum cuts are computationally complex. Metrics such as Sparsity and the Cheeger constant constitute combinatorial optimization problems. The Fiduccia–Mattheyses algorithm heuristically minimizes edge cuts under a given balance constraint in linear time and can be used to approximate the Cheeger constant. This algorithm is discussed in detail in the next section concerning optimization algorithms.

5.2.1 Evaluation Metric

The central aim of censorship resilience maximization is to diffuse control over information flow within a network. Increased path redundancy between node pairs reduces the influence of individual nodes or edges on communication. Accordingly, path redundancy in the SCION network is used to evaluate the optimization results. Specifically, the number of available SCION paths captures how many distinct communication paths the architecture provides. Unlike the previously defined objective functions, this metric is not purely analytical but depends on empirical measurements. Consequently, it is not used during optimization, but is instead evaluated after deploying a SCION topology to assess the performance of the optimization algorithms.

5.3 Optimization Algorithms

This section describes the optimization algorithms R_{AC} and R_{NP}. The former builds on the rewiring algorithm proposed in [31], adapting it to preserve SCION-specific constraints. The latter repeatedly applies the Fiduccia–Mattheyses partitioning algorithm [68] in a *divide-and-conquer* manner. The overarching algorithm R_{NP}, including its specific application of the pre-existing Fiduccia–Mattheysis partitioning, is entirely novel. The algorithms require a SCION topology G_{SCION} and an integer k as input. They return a modified G^*_{SCION}, which preserves the number of edges and vertices of G_{SCION}, with k edges having been rewired. Implementations of the algorithms are publicly available on GitHub [110] as part of the project discussed in Chapter 6.

5.3.1 Rewire for Algebraic Connectivity (R_{AC})

R_{AC} is based on an algorithm proposed by [31] that approximates optimal algebraic connectivity by using an upper and lower bound of μ_2. Let G be a graph with adjacency

spectrum $A(G)$, Laplacian matrix $L(G)$ and n vertices. Given an edge (i, j) to be removed from G, let $\mathbf{z} \in \mathbb{R}^n$ be a vector such that $z_i = 1$, $z_j = -1$, and all other entries are zero. In this work μ_i and $\mathbf{v}_i$ denote the i-th eigenvalue and eigenvector of $L(G)$, respectively. Let λ_i and $\mathbf{u}_i$ correspond to the eigenvalues and eigenvectors of the adjacency spectrum $A(G)$, respectively. The matrix $\Delta L = \mathbf{z}\mathbf{z}^T$ encodes the change in the Laplacian matrix $L(G)$, so that $L(G) - \Delta L = L(G - e_{ij})$. With this established, $\alpha_{ij} := |\langle \mathbf{z}, \mathbf{v}^{(2)}(G) \rangle| = |\mathbf{v}_i^{(2)}(G) - \mathbf{v}_j^{(2)}(G)|$.

In [31], the authors establish an upper and a lower bound for $\mu_2(G - e_{ij})$, both of which contain the term α. Therefore, α_{ij} can be used as an approximation for $\Delta\mu_2$ upon edge removal. Note that since α_{ij} is defined as an absolute value, it approximates the magnitude of the change in μ_2. Therefore, it can also be used for edge addition estimates. In the case of the edge rewiring application, an old edge (u, v) must have a minimal value $\alpha_{u,v}$ and a new edge (k, l) must have a maximal value $\alpha_{k,l}$.

This work puts forward a technical implementation of the algorithm proposed by [31]. Furthermore, the algorithm is adapted to SCION and extended to enforce its domain-specific constraints. The result is the algorithm R_{AC} shown in Algorithm 1. In the following paragraphs, helper structures are discussed, followed by a control flow description of R_{AC}.

In order to retrieve the minimal edge (u, v) and the maximal edge (k, l), all α values can be computed as an $n \times n$ matrix R, such that $R_{ij} = \alpha_{ij}$, using the following method:

$$R = \left|\mathbf{v}^{(2)}\mathbf{1}^T - \mathbf{1}(\mathbf{v}^{(2)})^T\right|$$

The pseudocode in Algorithm 2 contains the function corresponding to the construction of matrix R. Furthermore, a matrix C is constructed that encodes the SCION connectivity rules.

The algorithm starts by initializing the adjacency matrix $A(G)$. Additionally, the subgraph G_{CORE} induced by the set of core nodes is constructed and its matrix $A(G_{CORE})$ is derived. A look-up structure *core_map* which maps vertex indices from G to indices in G_{CORE}, is defined. Finally, the Laplacian matrix $L(G)$ is derived from $A(G)$ and μ_2 and $\mathbf{v}^{(2)}$ are computed by calculating the bottom-t eigenpairs of $L(G)$, with $t = 2$. Then, for each of the k iterations, the matrices C and R are derived, where k is a user-defined parameter denoting the number of edge rewirings.

The $n \times n$ matrix $R_{\max}$ is used to identify the highest-scoring non-existing edge (k, l) to be added to the graph. It is defined as $R_{\max} = R \cdot (C - A(G))$, thereby masking out both non-permissible and existing edges. Analogously, the $n \times n$ matrix $R_{\min}$ is used to identify the lowest-scoring existing edge to be removed. It is derived from R by masking out non-existing edges using $A(G)$. The algorithm then selects the corresponding maximal and minimal edges. The matrices $A(G)$ and $A(G_{CORE})$ are updated. Next, μ_2 and $\mathbf{v}^{(2)}$ and μ_2^{core} are calculated. If either μ_2 or μ_2^{core} are 0, indicating that the subgraph is disconnected, the entry (i, j) in R_{min} is masked out, and the while-loop repeats, selecting a new candidate for an edge (i, j), ensuring both global connectivity and core connectivity. If $\mu_2 > 0$ and $\mu_2^{core} > 0$, the edge rewiring is performed and the algorithm progresses to the next iteration.

Runtime Analysis

The outer loop runs k times. The construction of the matrices R, $R_{\max}$, $R_{\min}$, and C each requires $O(n^2)$ time. The inner loop may iterate over multiple candidate edges if removing a maximum-score edge would disconnect the core subgraph. However, this situation is unlikely, as the robustness objective inherently discourages disconnecting core nodes. Empirically, the while loop typically terminates after a single iteration and can therefore be treated as constant time. The retrieval of the minimal and maximal values in matrices R_{min} and R_{max} requires $O(n^2)$ time for each. The smallest two eigenpairs $L(G)$ and $L(G_{CORE})$ must be recalculated within each inner iteration. Assuming the use of an efficient method to calculate the smallest two eigenpairs of the sparse Laplacian matrices, such as the Lanczos algorithm [111], [112], μ_2 can be calculated in $O(m)$ per matrix. Combining the accumulated costs results in a total runtime of $O(k \cdot (m + n^2))$. For sparse graphs, with $m \ll n^2$, the term n^2 is dominant, resulting in a total complexity of $O(k \cdot n^2)$.

5.3.2 Rewire by Network Partition (R_{NP})

R_{NP} is a novel algorithm that repeatedly applies the network partitioning approximation proposed by Fiduccia and Mattheyses in [68] in a *divide-and-conquer* manner. The Fiduccia–Mattheyses (FM) algorithm is a min-cut heuristic that runs in linear time per pass with respect to the size of the graph and produces an approximate minimum partition subject to a balance constraint. In order to successfully apply it within R_{NP}, the FM algorithm is further modified in two ways: (1) It takes the additional argument S, where $S \subset V$, which functions as a node mask and allows the algorithm to operate on a subset of the graph. (2) When updating the current optimal cut, this version breaks ties by favoring the partition that has a smaller sum of degrees. Algorithm 3 presents the FM algorithm with the aforementioned modifications. Since R_{NP} uses the FM algorithm as an internal subroutine, the following paragraphs are distinguished between the subroutine and the overarching algorithm by marking them explicitly as "FM–R_{NP}" and "R_{NP}", respectively. The same notation holds for the captions of the algorithms.

FM–$\mathbf{R_{NP}}$: Initialization

The heuristic partition procedure works as follows. Given a graph G, a ratio r and a tolerance m, and a set of vertices S, a random partition A, B such that $A \cup B = V(G) \setminus S, \quad A \cap B = \emptyset$ is initialized. The partition is constructed to satisfy the balance constraint $|\frac{min(|A|,|B|)}{max(|A|,|B|)} - r| \leq m$, meaning that the ratio between the sizes of the two subsets approximates r within tolerance of m. Furthermore, nodes in the network can only be moved once and are rendered unmovable afterward. All nodes that are not in S are initially marked as movable.

FM–$\mathbf{R_{NP}}$: Loop

As a next step, the algorithm repeats a loop for as long as any node is movable. In each iteration, a gain function determines the most favorable move among the eligible nodes. The gain is defined so that greedily selecting the node with maximum gain reduces the

cut size. Thus, the gain of flipping node i is the resulting change in cut size and is defined as $g(i) = |\partial_i A||\bar{\partial}_i A|$, where $|\partial_i A|$ are the cut edges of i and $|\bar{\partial}_i A|$ its non-cut edges. In each iteration, the movable node with maximum gain moves to the opposite partition if the balance constraint allows. The overall minimum partition is updated if a more optimal cut is found. Ties are broken by favoring the partition with a smaller sum of node degrees, which helps counteract unfavorable cuts that may arise when a node has many edges connecting to the masked set S. This policy constitutes modification (2). An example of this tie breaking is shown in Figure 5.2.

FM–R_{NP}: Passes

The previously described initialization and loop procedure finds locally optimal cuts. In order to find a global optimum, it is repeated over several passes. The original FM algorithm requires only a small number of passes, as the algorithm tends to converge rapidly. In practice, convergence is often observed within at most four passes. However, in this work, 20 passes were used due to the modified application setting of the FM algorithm within R_{NP}.

In this context, not all minimum cuts are equally desirable. This is a consequence of missing structural information induced by masked nodes, which reduces the effective context available to the partitioning procedure. To address this, the tie-breaking heuristic was introduced that prefers cuts whose partition has a smaller total degree sum. The degree sum serves as an approximation for the missing connectivity information: although edges to masked nodes are excluded from the cut computation, their influence is still partially reflected in the node degrees. Prioritizing partitions with smaller total degree therefore biases the algorithm toward cuts that are more structurally significant in the global graph.

As shown in Algorithm 3, the modified FM algorithm returns an object np, containing the Cheeger constant np.cheeger and the selected partition $np.partition$.

R_{NP}: Retrieving the Old Edge

The algorithm R_{NP} uses the sub-procedure `Divide_And_Conquer_Max` shown in Algorithm 4 to identify a "maximally expendable edge". `Divide_And_Conquer_Max` takes as input a graph G, partition result np, a set of active nodes X, and masked nodes S, and proceeds as follows. After the global minimum network partition result np with partitions P_A, P_B is derived, it is passed to `Divide_And_Conquer_Max` with $X = V(G)$ and $S = \emptyset$. A further network partition is run on both P_A and P_B. Let P_{max} denote the partition among P_A and P_B with the larger Cheeger constant, and let P_{min} denote the partition with the smaller Cheeger constant. The set of active nodes is then overwritten $X \leftarrow P_{max}$ and the set of masked nodes is updated by $S \leftarrow S \cup P_{min}$. The algorithm then recurses on set X with masked nodes S. This procedure is illustrated in Figure 5.1. The algorithm terminates if $|X| = 2$, and the edge between the two nodes in X is returned as the maximally expendable edge. A case of termination is visualized in Figure 5.2, which also includes an example of a tie-break.

R_{NP}: Retrieving the New Edge

In order to determine a maximally beneficial new edge, the sub-procedure shown in Algorithm 5 recursively examines partitions, similarly to `Divide_And_Conquer_Max`. In

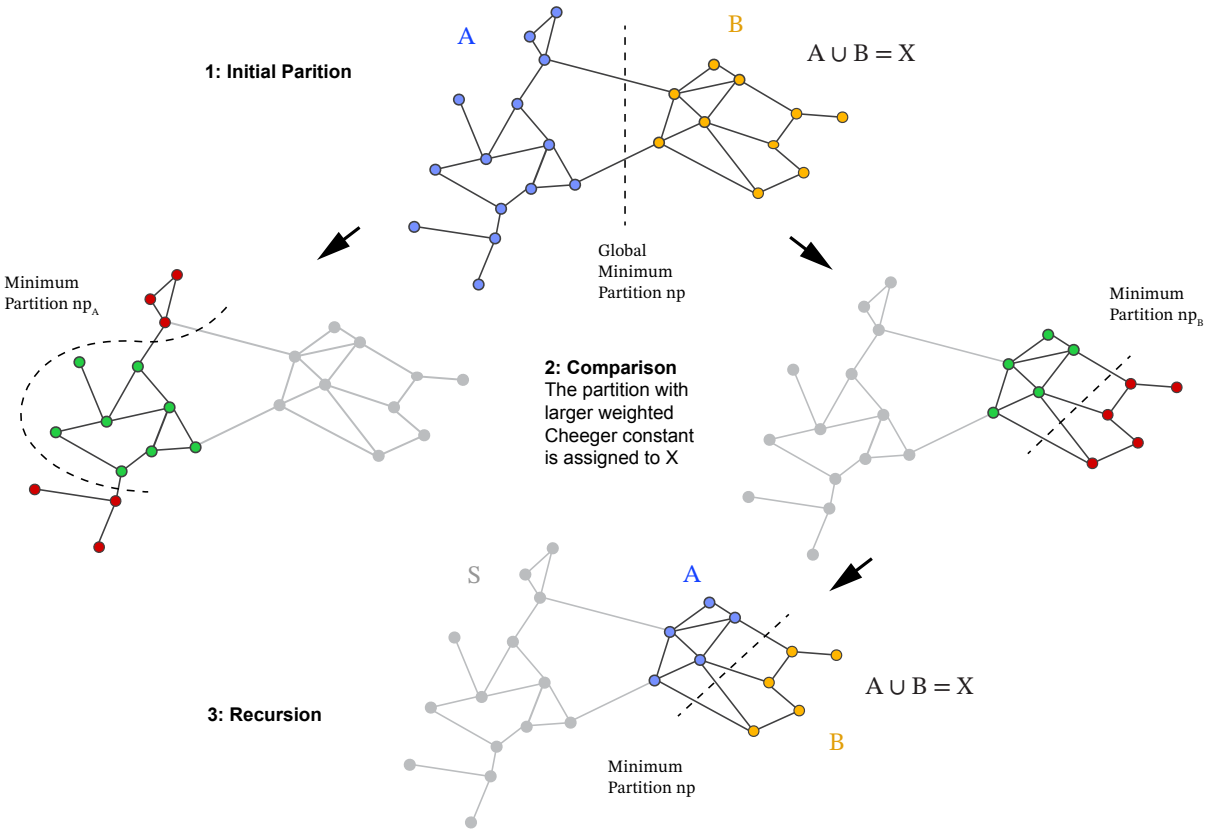

Figure 5.1: R_{NP}: `Divide_And_Conquer_Max` initial recursion step on example network.

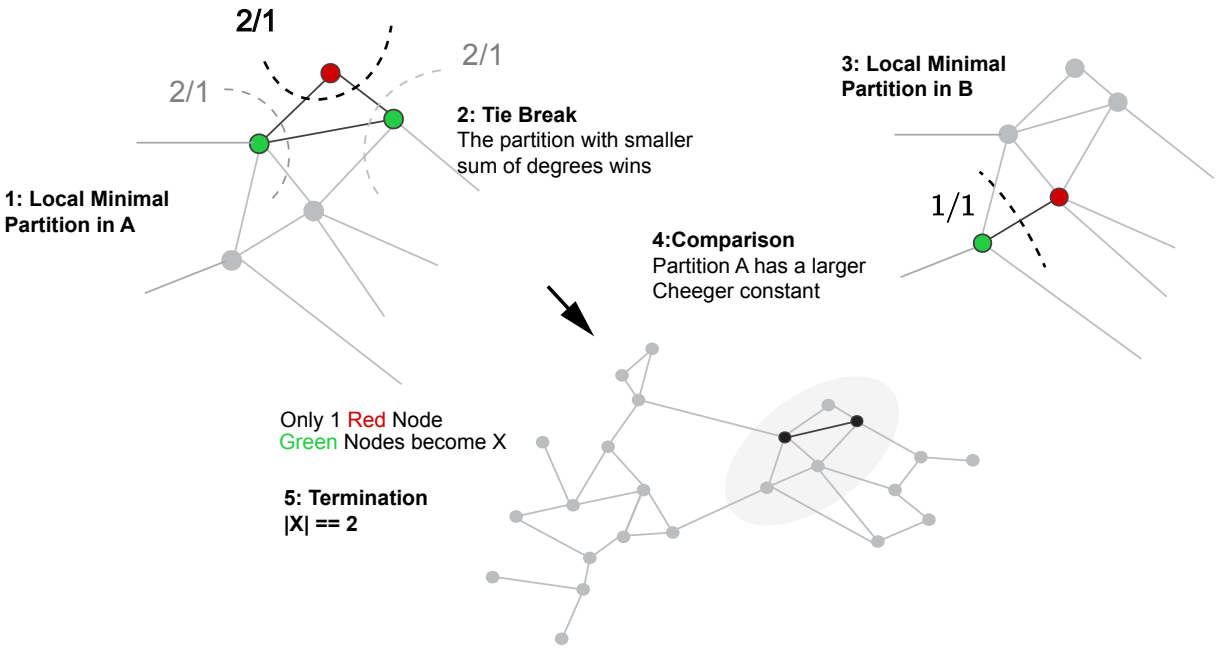

Figure 5.2: R_{NP}: `Divide_And_Conquer_Max` termination, including tie-break example.

contrast however, `Divide_And_Conquer_Min` recurses on the partition with the smaller Cheeger constant. With each iteration a counter is increased. In an additional data structure `node_scores`, nodes included in the current partition X are mapped to the current count. This procedure is run independently on each of the two partitions P_A and P_B of the global minimum cut. Finally, the nodes $u \in P_A$ and $v \in P_B$ with the highest score in their respective partitions are selected. Furthermore, in this step, the SCION-connectivity constraint (**C1**) is enforced using matrix C, which allows for only eligible node pairs.

The construction of the validity matrix is given by Algorithm 6. It takes as input the adjacency spectrum A, the partition P_A, the partition P_B and the number of nodes n. The matrix $W = C - A(G)$ encodes eligible and non-existing edges. In the construction of the matrix C, a further step ensures that u, v are of partitions P_A and P_B, respectively. This is achieved by encoding P_A and P_B in matrices U and V, where diagonal entries are 1, if they belong to the corresponding partition, and 0 otherwise. The construction of these matrices is denoted in Algorithm 6 as $\texttt{diag}(P_A, n)$ and $\texttt{diag}(P_B, n)$. Multiplying $U \cdot W \cdot V$ zeroes out entries where $i \notin P_A \vee j \notin P_B$. This ensures that any entry equal to 1 satisfies $i \in P_A \wedge j \in P_B$.

$\mathbf{R_{NP}}$: Main Procedure

Having described all the necessary sub-procedures of R_{NP}, the overarching logic, shown in Algorithm 7, works as follows. For each of the k iterations, a global minimum partition np with sets P_A and P_B is derived. This partition is then used to find the old edge and the new edge with the procedures described above. The changes are applied to the graph and np is recomputed. If the Cheeger value has decreased, the change is reverted. Furthermore, within the iteration, the algorithm enforces **C2**, ensuring that the core nodes form a connected subgraph. The vertex set $S_{\text{non-core}}$ contains all non-core nodes of the graph. The network partition can then be run with masked nodes $S_{\text{non-core}}$. If the Cheeger constant is 0, the subgraph is disconnected and the change is reverted. After k iterations, the modified graph G is returned.

Notes on Values r and m

The balance constraint of the FM algorithm ensures that the ratio of partitions P_A and P_B is approximately r. In order to find a global Cheeger constant, independent of the ratio r, the algorithm can be executed for values $[0.05, 0.15, 0.25, 0.35, 0.45]$, with $m = 0.05$ [43], hence covering all possible ratios. This configuration is shown in Algorithm 3. In R_{NP}, these values are used to derive the initial global partition, as well as the partition of the SCION core. In contrast, the balance constraint is employed differently within the *divide-and-conquer* sub-procedures. Here, values $[0.37524, 0.4575]$ are used for r, with $m = 0.0425$. With this configuration, the maximal ratio is 0.5, corresponding to the optimal split in terms of runtime. The minimum ratio is 0.33, which represents a compromise between runtime and flexibility, ensuring that node sets of three can still be processed.

Runtime Analysis

A single pass of the FM algorithm is $O(n)$ [68], with n being the size of the network. This yields an overall runtime of $O(r \cdot p \cdot n)$, where r denotes the number of parameter settings

considered and p the number of passes. If both r and p are treated as constants, this simplifies to $O(n)$.

However, as discussed earlier in this subsection, considering the optimal choice of minimum cuts, it is possible that a more effective implementation may require p to scale with the problem size, in which case the runtime of the FM algorithm would no longer be linear. Further investigation is required to confirm this.

Separately, it should be noted that the implementation of the FM algorithm used in the experiments exhibits a higher runtime complexity than $O(n)$, independent of p. This is due to specific implementation details and does not affect the theoretical analysis. In principle, the implementation could be adapted to match the FM algorithm's theoretical complexity.

The recursive *divide-and-conquer* algorithms divide the node set roughly in half with each iteration. Therefore, the expected recursion depth is $\log n$, resulting in a total recursive runtime of $O(n \log n)$. In each iteration, the validity matrix is constructed and scanned for a minimal entry, which takes $O(n^2)$ time. Therefore, the final runtime is $O(k \cdot (n^2 + n \log n))$, where k is the number of rewiring operations.

Input: Graph G, number of iterations k
Output: Modified graph G
$G_c \leftarrow \texttt{core}(G)$
$A \leftarrow \texttt{adjacency\_matrix}(G)$
$A_c \leftarrow \texttt{adjacency\_matrix}(G_c)$
core_map $\leftarrow \texttt{create\_core\_node\_map}(G)$
$(\mu, V) \leftarrow \texttt{get\_bottom\_t\_eigenpairs}(\texttt{adjacency\_to\_laplacian}(A), 2)$
for $i \leftarrow 1$ **to** k **do**
  flag $\leftarrow$ False
  $C \leftarrow \texttt{build\_validity\_matrix}(G, A)$
  $R \leftarrow \texttt{build\_alpha\_matrix}(V)$
  $R_{\max} \leftarrow R \cdot C$
  $R_{\min} \leftarrow R$
  foreach (u, v) **do**
    if $A[u, v] = 0$ **then**
      $R_{\min}[u, v] \leftarrow \infty$
    end
  end
  while *flag = False* **do**
    $(u_{\text{old}}, v_{\text{old}}) \leftarrow \arg\min_{(u,v)} R_{\min}[u, v]$
    $(u_{\text{new}}, v_{\text{new}}) \leftarrow \arg\min_{(u,v)} R_{\max}[u, v]$
    $A \leftarrow \texttt{update\_adjacency}(A, u_{\text{old}}, v_{\text{old}}, 0)$
    $A \leftarrow \texttt{update\_adjacency}(A, u_{\text{new}}, v_{\text{new}}, 1)$
    foreach $(u, v, x) \in \{(u_{old}, v_{old}, 0), (u_{new}, v_{new}, 1)\}$ **do**
      if $\{u, v\} \subseteq$ *core_map* **then**
        $A_c \leftarrow \texttt{update\_adjacency}(A_c, \text{core\_map}[u], \text{core\_map}[v], x)$
      end
    end
    $(\mu, V) \leftarrow \texttt{get\_bottom\_t\_eigenpairs}(\texttt{adjacency\_to\_laplacian}(A), 2)$
    $(\mu_c, V_c) \leftarrow \texttt{get\_bottom\_t\_eigenpairs}(\texttt{adjacency\_to\_laplacian}(A_c), 2)$
    if $\mu_2 \leq 0$ ***or*** $\mu_{c,2} \leq 0$ **then**
      $R_{\min}[u_{\text{old}}, v_{\text{old}}] \leftarrow \infty$
      continue
    end
    $G \leftarrow G - (u_{\text{old}}, v_{\text{old}}) + (u_{\text{new}}, v_{\text{new}})$
    flag $\leftarrow$ True
  end
end
return G

Algorithm 1: R_{AC}: The algebraic connectivity rewiring strategy.

Function `build_alpha_matrix(`V`):`
  $v_2 \leftarrow$ second eigenvector in V
  return $\left|v_2^2 + (v_2^2)^\top - 2\,(v_2 v_2^\top)\right|$
Function `build_validity_matrix(`G, A`):`
  $C_{uv} \leftarrow \begin{cases} 1 & \text{if } (u,v) \text{ are core nodes or } \mathrm{isd}(u) = \mathrm{isd}(v) \\ 0 & \text{otherwise} \end{cases}$
  return $C - A$
Function `update_adjacency(`A, u, v, x`):`
  $A[u,v] \leftarrow x$
  $A[v,u] \leftarrow x$
  return A

Algorithm 2: R_{AC}: Helper functions, including construction of validity matrix.

Input: Graph $G = (V, E)$, masked nodes $S \subseteq V$
Output: Best partition np
$R \leftarrow [0.05, 0.15, 0.25, 0.35, 0.45]$;
$m \leftarrow 0.05$;
cheeger $\leftarrow \infty$;
$P_{A^*} \leftarrow$ Nil;
for *each ratio* r *in* R **do**
  for $i \leftarrow 1$ **to** 20 **do**
    $(P_A, P_B) \leftarrow$ FM_Partition(G, r, m, S);
    cur_cheeger $\leftarrow \frac{\text{nr\_cut\_edges}}{\min(|P_A|,|P_B|)}$;
    `// Update if strictly better, or tied but smaller sum of degrees`
    if *cur_cheeger* $\leq$ *cheeger* $\vee$ ((*cur_cheeger* $=$ *cheeger*) $\wedge$ (`degree_sum`$(P_A) \leq$ `degree_sum`$(P_{A^*}))$) **then**
      cheeger $\leftarrow$ cur_cheeger;
      $P_{A^*} \leftarrow P_A$;
    end
  end
end
$np \leftarrow \{\text{cheeger} : \text{cheeger},\ \text{partition} : P_{A^*}\}$;
return np;

Function *BalanceOK*$(|P_A|, |P_B|, r, m)$**:**
  `//` P_A, P_B `contain only non-masked nodes`
  $n \leftarrow |P_A| + |P_B|$;
  ratio $\leftarrow \frac{min(|P_A|,|P_B|)}{n}$;
  if $|ratio - r| \leq m$ **then**
    return TRUE;
  end
  return FALSE;

Function *FM_Partition*(G, r, m, S)**:**
  Initialize balanced partition (P_A, P_B) over $V(G) \setminus S$ according to r;
  Assign all $v \in S$ to a fixed dummy partition, locked permanently;
  Compute gains for all $v \in V(G) \setminus S$;
  while *unlocked vertices in* $V(G) \setminus S$ *exist* **do**
    $v \leftarrow$ non-masked vertex with maximum gain;
    if *BalanceOK*$(|P_A|', |P_B|', r, m)$ *after moving* v **then**
      Move v to opposite partition;
      Lock v;
      Update gains of non-masked neighbors of v;
    else
      Find v with highest gain for which balance is OK;
    end
  end
  return (P_A, P_B);

Algorithm 3: FM–R_{NP}: Fiduccia-Mattheyses network partitioning with modifications.

Function `divide_and_conquer_max`(G, np, X, S)**:**
  if np.cheeger $= 0$ **then**
    return Nil;
  end
  if $|X| = 2$ **then**
    Let (u, v) be the two nodes in X;
    return (u, v);
  end
  $P_A \leftarrow np$.partition;
  $P_B \leftarrow X \setminus P_A$;
  if $|P_A| = 1$ **then**
    $np_b \leftarrow$ `network_partition`$(S \cup P_A)$;
    return `divide_and_conquer_max`$(G, np_b, P_B, S \cup P_A)$;
  else if $|P_B| = 1$ **then**
    $np_a \leftarrow$ `network_partition`$(S \cup P_B)$;
    return `divide_and_conquer_max`$(G, np_a, P_A, S \cup P_B)$;
  end
  $np_a \leftarrow$ `network_partition`$(S \cup P_B)$;
  $np_b \leftarrow$ `network_partition`$(S \cup P_A)$;
  if np_a.cheeger $> np_b$.cheeger $\vee$ $((np_a$.cheeger $= np_b$.cheeger$) \wedge$ (`degree_sum`$(np_a) \leq$ `degree_sum`$(np_b))$) **then**
    return `divide_and_conquer_max`$(G, np_a, P_A, S \cup P_B)$;
  else
    return `divide_and_conquer_max`$(G, np_b, P_B, S \cup P_A)$;
  end

Algorithm 4: R_{NP}: The `Divide_And_Conquer_Max` sub-algorithm procedure.

Function `find_new_edge`$(G, np, X, scores)$**:**
  $P_A \leftarrow np$.partition;
  $P_B \leftarrow X \setminus P_A$;
  $validity\_mat \leftarrow$ `build_validity_matrix_2`$(A(G), P_A, P_B, |V(G)|)$;
  $score \leftarrow$ `get_new_edge_scores`(G, np);
  $u, v \leftarrow \text{argmax}(score)$ with $validity\_mat_{u,v} = 1$;
  return (u, v);

Function `get_new_edge_scores`(G, np)**:**
  $P_A \leftarrow np$.partition;
  $P_B \leftarrow V(G) \setminus P_A$;
  $node\_scores \leftarrow \text{ZeroArray}(|V(G)|)$;
  if $|P_A| > 1$ **then**
    $np_a \leftarrow$ `network_partition`(G, P_B);
    `divide_and_conquer_min`$(G, np_a, P_B, node\_scores, 0)$;
  end
  if $|P_B| > 1$ **then**
    $np_b \leftarrow$ `network_partition`(G, P_A);
    `divide_and_conquer_min`$(G, np_b, P_A, node\_scores, 0)$;
  end
  return $node\_scores$;

Function `divide_and_conquer_min`$(G, np, S, node\_scores, counter)$**:**
  $P_A \leftarrow np$.partition;
  $P_B \leftarrow V(G) \setminus (S \cup P_A)$;
  foreach $u \in P_A$ **do**
    $node\_scores[u] \leftarrow counter$;
  end
  foreach $v \in P_B$ **do**
    $node\_scores[v] \leftarrow counter$;
  end
  $counter \leftarrow counter + 1$;
  if $|P_A| \leq 1$ ***or*** $|P_B| \leq 1$ **then**
    return;
  end
  $np_a \leftarrow$ `network_partition`$(G, S \cup P_B)$;
  $np_b \leftarrow$ `network_partition`$(G, S \cup P_A)$;
  if np_a.cheeger $< np_b$.cheeger $\vee$ ((np_a.cheeger $= np_b$.cheeger) $\wedge$ (`degree_sum`$(np_a) \leq$ `degree_sum`(np_b))) **then**
    `divide_and_conquer_min`$(G, np_a, S \cup P_B, node\_scores, counter)$;
  else
    `divide_and_conquer_min`$(G, np_b, S \cup P_A, node\_scores, counter)$;
  end

Algorithm 5: R_{NP}: `Divide_And_Conquer_Min` with retrieval of the new edge.

Function `build_validity_matrix_2`(A, P_A, P_B, n)**:**

$C_{uv} \leftarrow \begin{cases} 1 & \text{if } u, v \text{ are core nodes or } \mathrm{isd}(u) = \mathrm{isd}(v) \\ 0 & \text{otherwise} \end{cases}$

$W \leftarrow C - A$

$U \leftarrow \mathrm{diag}(P_A, n)$

$V \leftarrow \mathrm{diag}(P_B, n)$

return $U \cdot W \cdot V$

Algorithm 6: R_{NP}: Construction of validity matrix.

Input: Graph G, iterations k

for $i \leftarrow 1$ **to** k **do**

  $np \leftarrow$ `network_partition`$(G, \emptyset)$;

  $non\_core\_nodes \leftarrow V(G) \setminus$ `core(`G`)`;

  $del\_edge \leftarrow$ `divide_and_conquer_max`$(G, np, G.nodes, \emptyset)$;

  $node\_scores \leftarrow$ `get_new_edge_scores`(np);

  $new\_edge \leftarrow$ `find_new_edge`$(np, G.nodes, node\_scores)$;

  if $del\_edge \neq \emptyset$ ***and*** $new\_edge \neq \emptyset$ **then**

    $H \leftarrow G - del\_edge + new\_edge$;

    $np_H \leftarrow$ `network_partition`$(H, \emptyset)$;

    $np_{core} \leftarrow$ `network_partition`(H, non_core_nodes);

    if $np_H.cheeger < np.cheeger$ **then**

      continue;

    end

    if $np_{core}.cheeger \leq 0$ **then**

      continue;

    end

    $G \leftarrow H$;

    $np \leftarrow np_H$;

  end

end

Algorithm 7: R_{NP}: The main procedure.

Chapter 6

Implementation

In this chapter, the implementation of a configurable SCION testbed is discussed. In order to test and compare the censorship resilience of different topologies, a testbed must facilitate a simple workflow of reconfiguration, deployment, and evaluation.

This implementation is based on the testbed project published in [113] and extends it with dynamic configurability. The resulting testbed implementation is publicly available on GitHub [110].

This project is compatible with SCION version 0.14.0 [114] and the results provided in Chapter 7 were produced using this version.

The rest of this chapter is organized as follows. First, the initial architecture is briefly described, establishing the point of departure. Next, points of necessary intervention are discussed, followed by the respective modification. Then, a summary of the final architecture and deployment logic is presented. Finally, the automated optimization and evaluation pipelines are discussed.

6.1 Initial Architecture

This section covers the most relevant aspects of the implementation as designed in [113] is presented. Originally based on the Freestanding SCION Network tutorial [1], a testbed was developed in which each SCION AS is simulated with a Docker container.

The folder structure of the testbed is organized as follows. Some entries have been omitted in accordance with relevance.

- `/base/`
  - `Dockerfile` from which all SCION nodes inherit. It downloads SCION binaries, initializes the web server, and executes the PKI generation scripts.

[1] https://docs.scion.org/en/latest/tutorials/deploy.html

  - `/pki/` contains four bash scripts that generate the public keys for each ISD. Additionally, these scripts generate certificates for three core nodes, where AS1 and AS3 act as voting nodes (required for Trust Root Configuration), and AS1 and AS2 act as certificate authorities (CA).

- `/ISD1-ISD4/` For each ISD, there is one directory containing five ASes. The subdirectories are named `scion[ISD Number][AS Number]`. Each AS directory contains:
  - an AS-specific `Dockerfile`
  - `topology[index].json`, which defines the SCION node's view of the network topology, including links, ports, addresses, and relationship types (peer, provider, or customer)

- `/monitor/`
  - `Dockerfile` for the monitoring node used during deployment
  - `scionctl`, a CLI tool for interacting with SCION nodes running in Docker containers

- `/test/` contains `scion_bat_test.bats` and `scion_ping_test.bats` for validating functionality after deployment.

- `docker-compose.yml` lists all Docker containers which are deployed, configures communication network across nodes, and specifies which ports are exposed and which volumes are used.

- `docker-compose.mac.yml` provides platform-specific overrides, in this case of the volume configuration when deployed on Mac.

- `Makefile` contains the command to build all SCION nodes for each ISD and deploy the network.

Limitations in Configurability

While this architecture supports straightforward deployment of a Docker-based SCION network with four ISDs, the network structure is tightly embedded in the implementation. Implementation details that prohibit changes in topology include the four `pki-generation` scripts. These are limiting in that they allow strictly four ISDs, while also hard-coding three core nodes with specific roles into the ISD topology. The scripts differ only slightly from one another, presenting an opportunity for automation.

Furthermore, the architecture contains one hard-coded folder per ISD, with five ASes as subfolders. The Dockerfiles contained in each of these directories also vary little across ASes and ISDs and could be parameterized. While the aforementioned design choices fix the number of ISDs and ASes that can be deployed, the `topology[index].json` files determine the edges between nodes. To modify how ASes are connected, these files must be updated.

In addition, the test cases in `scion_bat_test.bats` and `scion_ping_test.bats` hinder testing of alternative topologies, as they hard-code queries to specific nodes.

Finally, the `docker-compose.yml` and `docker-compose.mac.yml` files hard-code which nodes are deployed.

6.2 Modifications

This section presents the modifications made to the architecture to support configurable topologies.

Network Configuration Format

To enable a configurable architecture, a YAML-based file format was defined, which consists of two parts:

- **ISDs:** Lists the ISDs in the network, with each entry specifying the total number of nodes (`n`) and the set of core nodes.
- **Topology:** Lists all edges in the network, independent of ISDs.

This file format supports an arbitrary number of ASes, ISDs, and core nodes. Furthermore, it defines the connections between nodes, providing a single source of truth for network deployment. The helper script `parse_topology.py` converts YAML files to and from annotated NetworkX [108] graphs.

Makefile

The Makefile in the root of the project serves as the entry point to the project. Additionally, it manages the execution of automation scripts and the passing of input files. The script `parse_isd_config.py` takes network configuration files discussed previously and produces a list of Makefile variables: one for the list of ISDs and a further variable for each ISD, storing its range of ASes. These variables are saved in `.isd-vars.mk`, where they are accessible to the Makefile. Using this information, the Makefile runs the SCION node build loop and passes the correct parameters to each Dockerfile, allowing further parametrization within the Dockerfiles.

Template Dockerfile

The original implementation relied on a rigid folder structure with one folder per ISD and one subfolder per SCION node, each containing an almost identical Dockerfile. These Dockerfiles have been replaced by `/template/Dockerfile`, which accepts parameters `ISD`, `AS`, and `INDEX`, rendering the former file structure redundant. The parameters are passed to the Dockerfile by the Makefile.

Naming Scheme and Addressing

The previous implementation of the testbed followed a naming scheme `scion[ISD Number][AS Number]`. This posed an immediate issue, as the naming logic is coupled to

single-digit topologies. For flexible testing, more than 9 nodes per ISD are likely necessary. Therefore, the naming scheme was changed to `scion[ISD Number]-[AS Number]`.

In addition to the names, the addressing schemes also relied on single digit encoding, with SCION nodes initially being encoded as `ffaa:1:[ISD Number][AS Number]`, which was modified to `ffaa:[ISD Number]:[AS Number]`. Furthermore, SCION Docker nodes require two networks: their own AS network, simulating an autonomous system structure that may contain end hosts, and a shared transit network, representing inter-domain routing between ASes. Initially, the transit addresses relied on the following single digit encoding: `10.100.0.[ISD Number][AS Number]`. By extending the address space, the addressing could be changed to `10.100.[ISD Number].[AS Number]`, which, taking into account standard network address reservations, caps the number of ASes per ISD at 254.

The script `generate_compose.py` takes a network configuration file as input and generates `docker-compose.yml` and `docker-compose.mac.yml`, defining the corresponding Docker nodes and the required networks.

Generating Topology Files

In order to allow for arbitrary edge configurations, the hard-coded connections in the `topology[index].json` files must be replaced. Each SCION node has its own file, representing its local view of the network. The script `generate_topologies.py` takes a network configuration file as input and generates each of the required files, whereby it ensures correct naming scheme, manages addressing and configures ports, ensuring they are non-overlapping and unique. The port scheme caps the number of nodes per AS at 99.

In addition to the aforementioned responsibilities, `generate_topologies.py` fulfills a further requirement. The resulting `topology.json` files specify, for each connection, whether it is a peer, customer, or provider connection. As discussed in Chapter 5, ASes form customer-provider relationships. The network configuration files contain simple, undirected edge lists and are not hierarchical. Thus, `generate_topologies.py` infers a hierarchy as follows: Firstly, all core nodes form peer relationships among themselves. Secondly, core nodes always form provider-customer relationships with their adjacent non-core ASes. For each non-core node, its level in the hierarchy is determined by its distance to the nearest core node. Nodes closer to the core form provider-customer relationships with adjacent nodes at a longer core distance, while adjacent nodes at equal distances form peer relationships. Core distances are derived using a multi-source breadth-first search. Through this inference step, network configurations files still provide a simple and flexible interface, while correct SCION hierarchies are derived under the hood.

Generating Keys and Certificates

While the original implementation relied on four almost identical bash scripts to generate keys and certificates, the script `pki_generation.py` replaces all of them. It dynamically generates keys and certificates for the specified number of ASes. Furthermore, the script parses the network configuration file to retrieve the ISD core nodes. To simplify the logic, each core node is assigned the same rights and responsibilities. Hence, `pki_generation.py` designates all core nodes as voting nodes and certificate authorities.

This allows for further flexibility in the configuration, as the number of cores per ISD can now range from 1 to the total number of nodes in the ISD.

Monitor

The monitor container includes the *scionctl* CLI tool, which had to be modified to support dynamic topologies. The tool relied on `nodeconfig.yaml` to map nodes to addresses, which is now generated automatically by `generate_node_config.py`. Furthermore, several components in the CLI tool had to be adapted to the renewed naming scheme.

Additionally, the monitor container also includes a server that provides a web-based UI for node monitoring. Previously, node names were hard-coded in the `index.html` file. To address this, an `index_template.html` file was introduced, which is now used by `generate_web_ui.py` to generate the `index.html`. The script `web-ui.go` was changed to serve the new index directly from the working directory, instead of embedding a static version.

Generating Tests

Finally, as the nodes in the topology change, the testing suite should be able to accurately assess the functionality of the current configuration. To achieve this, the script `generate_tests.py` automatically generates `scion_bat_test.bats` and `scion_ping_test.bats`, ensuring that a wide combination of both intra-ISD and inter-ISD communication tests are covered.

Having described the necessary components to support configurable topologies, the next section presents the final logic of the testbed.

6.3 Final Build Workflow

This section describes the final internal procedure for building and deploying an arbitrary topology.

Entry Point: `make up`

This command orchestrates the complete process of building and deploying a topology.

- Run `parse-isd-config.py`, generating ISD variables in `.isd-vars.mk`
- Include variables from `.isd-vars.mk` in `Makefile`
- Set `NETWORK_CONFIG` in `Makefile`, which stores the path to the current network configuration file
- **Build phase:**
  - `generate-compose`
  - `generate-topologies`

    - Generates `topology.json` files for the Docker containers
    - Stores them in `/tmp/container-topologies/`
  - `build-base`, building `/base/Dockerfile`, which:
    - Downloads dependencies and prepares the container filesystem
    - Copies the network configuration file into `/tmp/`
    - Runs `pki_generation.py`, which:
      - Parses ISDs, AS counts, and core nodes
      - Generates keys and certificates for all nodes
  - `build-monitor`
    - Runs `generate-nodeconfig` and `generate-web-ui`
    - Builds `/monitor/Dockerfile`
  - `build-scion`
    - Builds one container per AS using `/template/Dockerfile`
    - Parameters `ISD`, `AS`, and `INDEX` are passed from the `Makefile`
    - The image:
      - Inherits from the base image
      - Retrieves keys and certificates using `INDEX`
      - Loads the corresponding topology file
  - `generate-tests`
- Deploy the containers

6.4 Automated Topology Optimization and Testing

This section explains the process used to generate optimized topologies using the algorithms described in Chapter 5. The testbed includes a workflow that supports the automated generation of optimized topologies, including automatic evaluation and the generation of figures. As before, this process is orchestrated by the `Makefile`.

Entry Point: `make run-topology-optimizer`

This command runs an iterative optimization process on a set of initial topologies, comprising the following sub-commands.

- `topo-optim` runs the optimization algorithms for each initial topology (given as a network configuration file) for k iterations. The algorithms are executed by scripts `rewire_spectral.py` in the case of R_{AC} and `rewire_np.py` for R_{NP}. The scripts output new network configuration files for each iteration.
- `topo-eval` runs robustness metrics on each of the produced network configuration files and saves the results into a csv-file.

- `topo-plot` parses the CSV file and generates plots. Furthermore, it produces graph visualizations for each iteration of the optimization process.

Additionally, the structure of the `Makefile` allows each of the sub-commands to be executed independently.

In a subsequent step, the effective impact of the optimization on the SCION architecture is evaluated. This requires deploying SCION topologies generated by the algorithms and measuring the number of paths produced by the architecture. Since the previously described procedure generates multiple topologies, an automated workflow for deployment and path evaluation is required. The command `run-path-evaluation` addresses this.

Entry Point: `make run-path-evaluation`

- Each network configuration produced by the optimization algorithms is built and deployed sequentially.
  - After each deployment, the script waits for 30 seconds to allow SCION path beaconing to initialize.
  - While the network is running, the `show-paths` command executes SCION commands to list paths between selected node pairs. The output is stored in `topology_optimization/data`.
- `eval-paths` processes the raw data by extracting the number of paths and distinguishing between intra-ISD and inter-ISD paths. The results are written to a CSV file.
- `plot-paths` visualizes the results from the CSV file and saves the generated figures.

This implementation supports the automated evaluation of all topologies through an iterative deployment queue. Overall, the testbed enables configurable topologies, automated topology optimization, as well as automated deployment and evaluation.

Chapter 7

Evaluation

This chapter presents an evaluation of the optimization process and its effects on the SCION architecture. In these experiments, 12 initial topologies are optimized by the algorithms R_{AC} and R_{NP}, presented in Chapter 5. Each topology undergoes 5 iterations for each algorithm, resulting in 10 derived networks per initial topology and a total of 132 topologies.

The resulting topologies are then evaluated using three metrics: algebraic connectivity, the Cheeger constant, and the Border Breadth. Subsequently, each of the 132 topologies is deployed in the testbed, using the automated deployment pipeline described in Chapter 6.

The remainder of this chapter is structured as follows. First, the experimental setup is discussed and the initial topologies are described. Next, the results of the experiments are presented, starting with the robustness evaluation, followed by the results of testbed path experiments. Finally, the findings are discussed.

7.1 Experimental Setup

This section presents the experimental setup. Most importantly, this involves discussing the twelve baseline topologies, their properties, and their respective sources. The SCION topologies were obtained from three sources. The first represents the initial configuration of the SCION testbed. The second corresponds to the real-world SCION network SCIERA [8]. The remaining 10 networks were synthetically generated using a process outlined later in this section.

The SCIERA network [8] is a real-world SCION network connecting academic institutions world-wide. The topology data was obtained from the latest available SCIERA documentation [21] (accessed March 27, 2026). Figure 7.1 shows the retrieved topology, annotated with their corresponding labels assigned by the testbed. Note that for the optimization experiments, the edges marked as “under construction” were included. The final graph used in the experiments is shown in Figure 7.2.

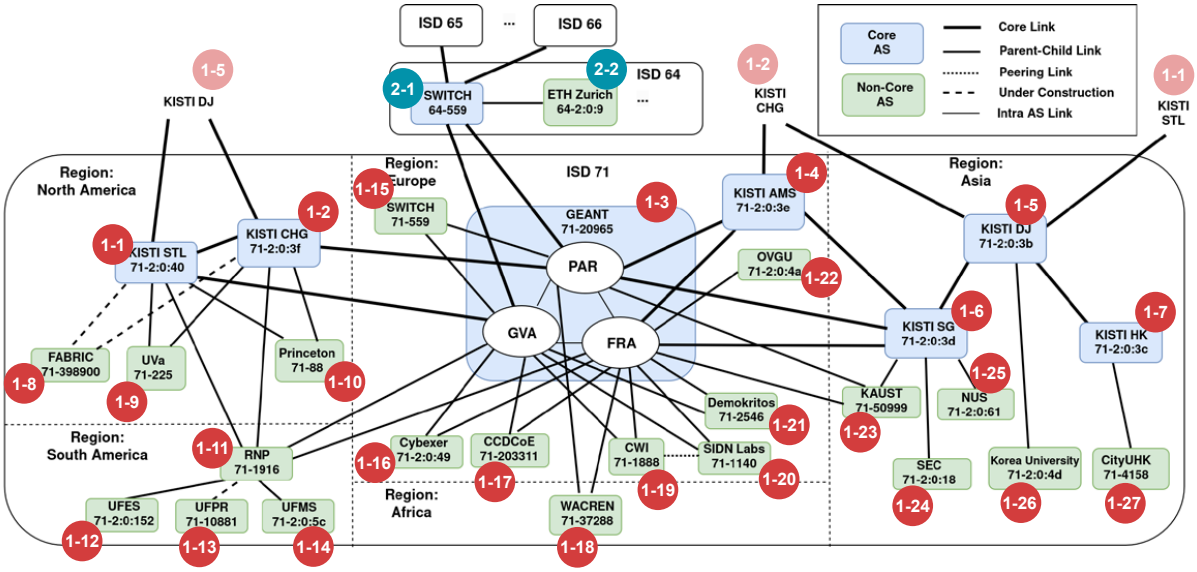

Figure 7.1: SCIERA topology, adapted from SCIERA documentation [21] (accessed March 27, 2026). The figure is extended with red labels indicating node indices used in subsequent graph representations.

Table 7.1 lists the 12 initial topologies, together with their number of nodes and edges, average degree, and number of ISDs and core ASes.

Topology	Nodes	Edges	Avg. Degree	# ISDs	# Core ASes
Testbed Topology	20	32	3.2	4	12
SCIERA Topology	29	38	2.62	2	8
Synthetic Topology 1	34	41	2.41	6	15
Synthetic Topology 2	49	65	2.65	4	8
Synthetic Topology 3	48	68	2.83	6	16
Synthetic Topology 4	42	54	2.57	5	9
Synthetic Topology 5	45	71	3.16	3	14
Synthetic Topology 6	40	65	3.25	3	13
Synthetic Topology 7	35	68	3.89	5	19
Synthetic Topology 8	23	39	3.39	3	9
Synthetic Topology 9	38	74	3.89	3	19
Synthetic Topology 10	33	42	2.55	5	10

Table 7.1: Topology statistics

7.1.1 Generating SCION Networks

To produce more generalizable experiments, synthetic SCION networks were required. Generating a SCION network is non-trivial. This section outlines the process used to generate such networks. Furthermore, the process is summarized in Algorithm 8.

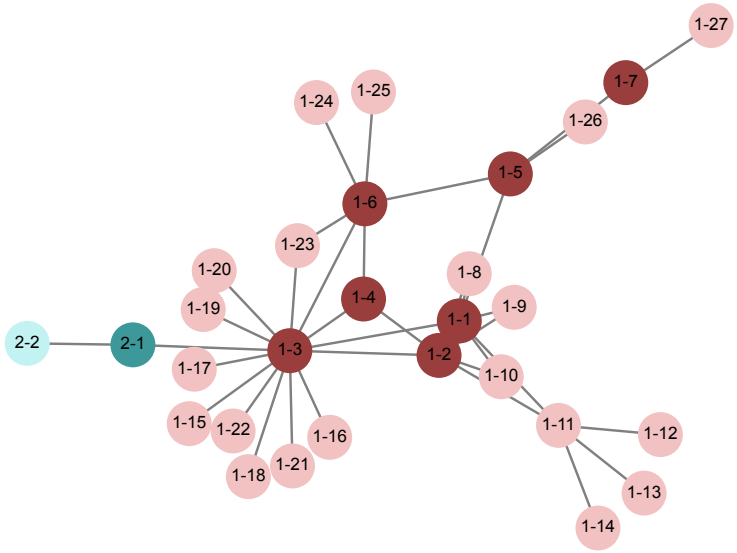

Figure 7.2: Initial SCIERA topology.

To obtain realistic network structures that reflect the scale-free nature of the Internet, the synthesis is based on the Barabási-Albert (BA) model, which produces graphs with a power-law degree distribution. The BA method takes as input the number of nodes n and a parameter m, which determines how many edges each newly added node forms.

Using this graph generation method, networks are initially generated with $m = 3$ and $n \sim \text{uniform}(20, 50)$. While a BA network with $m = 3$ results in a relatively dense graph, a second step partitions the network into ISDs, thereby sparsifying it.

The partitioning procedure works as follows. First, for each ISD, a random initial core node is selected. The ISDs are then expanded in a round-robin fashion: at each step, the current ISD incorporates a random node among its current neighbors. Once all ISDs have reached their assigned size, the remaining core nodes are selected randomly from the global network.

Finally, all edges that connect non-core nodes across different ISDs are deleted, which constitutes the aforementioned sparsifying step. If the resulting core subgraph is not connected, additional edges are inserted randomly to ensure connectivity.

This procedure aims to preserve the structural properties of a scale-free network while enforcing SCION-specific constraints. The 10 synthetic networks generated using this method are listed in Table 7.1.

With the initial topologies defined, the following section presents the experimental results.

7.2 Results

During the process of the experiments, the algorithms R_{AC} and R_{NP} were applied to the 12 baseline topologies. Each of the algorithms performed five edge rewirings. Resulting

$n \leftarrow \text{Uniform}(20, 50)$;
$n_{\text{ISD}} \leftarrow \text{Uniform}(3, 6)$;
$n_{\text{core}} \leftarrow \max(\text{Uniform}(6, 20),\ n_{\text{ISD}})$;
$G \leftarrow \text{BarabasiAlbert}(n,\ m = 3)$;
Mark all nodes as non-core;
$(s_1, \ldots, s_{n_{\text{ISD}}}) \leftarrow$ split n into n_{ISD} group sizes, each of size ≥ 3;
for *each ISD* i **do**
  Assign a random unassigned node to ISD i and mark it as core;
end
while *any ISD* i *has fewer than* s_i *assigned nodes* **do**
  for *each unfilled ISD* i **do**
    Expand ISD i by assigning a random neighbor of its current nodes;
  end
end
Randomly promote nodes to core until n_{core} core nodes are reached;
Remove all edges between non-core nodes that span different ISDs;
$G_{\text{core}} \leftarrow$ subgraph of G induced by core nodes;
if G_{core} *is disconnected* **then**
  Connect components by adding one edge between a random node from each;
end

Algorithm 8: Synthetic SCION network generation.

topologies were saved in both their final state and in each intermediate step. Figure 7.3 shows the result of R_{AC} and R_{NP} after performing five iterations on the baseline SCIERA topology shown in Figure 7.2.

The remainder of this section first presents the evaluation of the generated topologies using robustness metrics, and then the experimental evaluation conducted on the SCION testbed.

7.2.1 Robustness Metrics

In this section, the effect of the optimization algorithms R_{AC} and R_{NP} on the initial topologies is evaluated with regard to algebraic connectivity, the Cheeger constant and the Border Breadth.

Figure 7.4 shows the algebraic connectivity $a(G)$ of each baseline topology, as both R_{AC} and R_{NP} perform five edge swaps. Each plot includes the initial state at step 0. Considering the results, it is apparent that R_{AC} outperforms R_{NP} in this context by a wide margin. One can further observe that despite the good performance of R_{AC}, the results do not increase monotonically. This shows that the procedure using the bounds of $a(G)$ is approximate. R_{NP} performs poorly overall with regard to algebraic connectivity, but still produces an increase in most cases. In the case of the Testbed Topology, it even significantly outperforms R_{AC}.

Figure 7.5 shows the effect of the topology optimization on the Cheeger constant. For

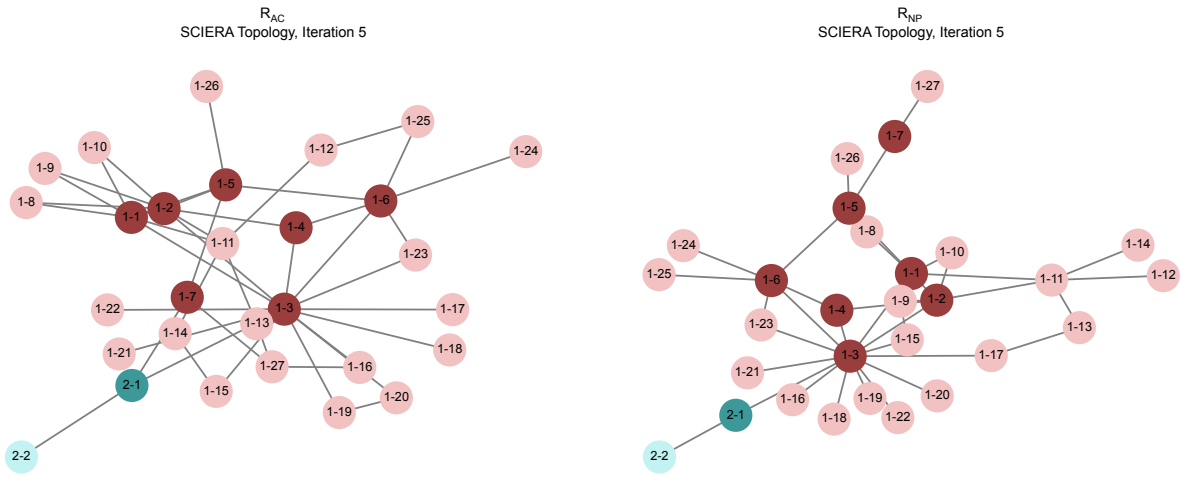

Figure 7.3: Final SCIERA topologies generated by R_{AC} and R_{NP}.

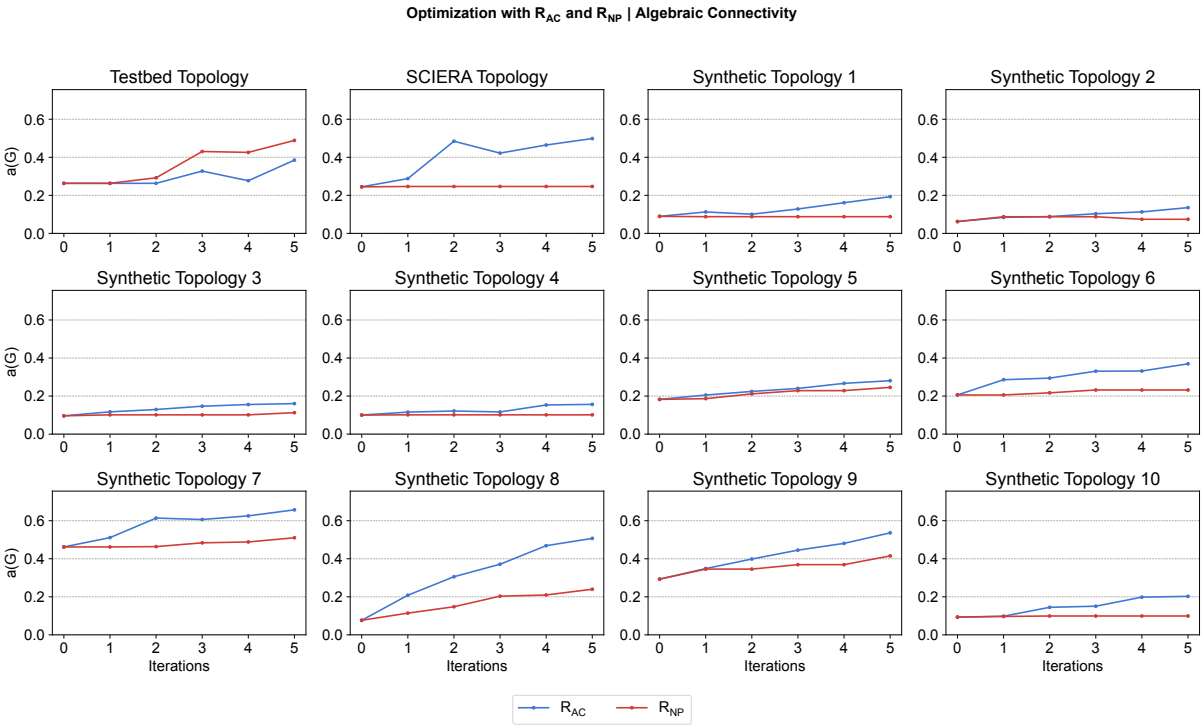

Figure 7.4: Algebraic connectivity $a(G)$ over iterations of optimization algorithms R_{AC} and R_{NP} for all topologies.

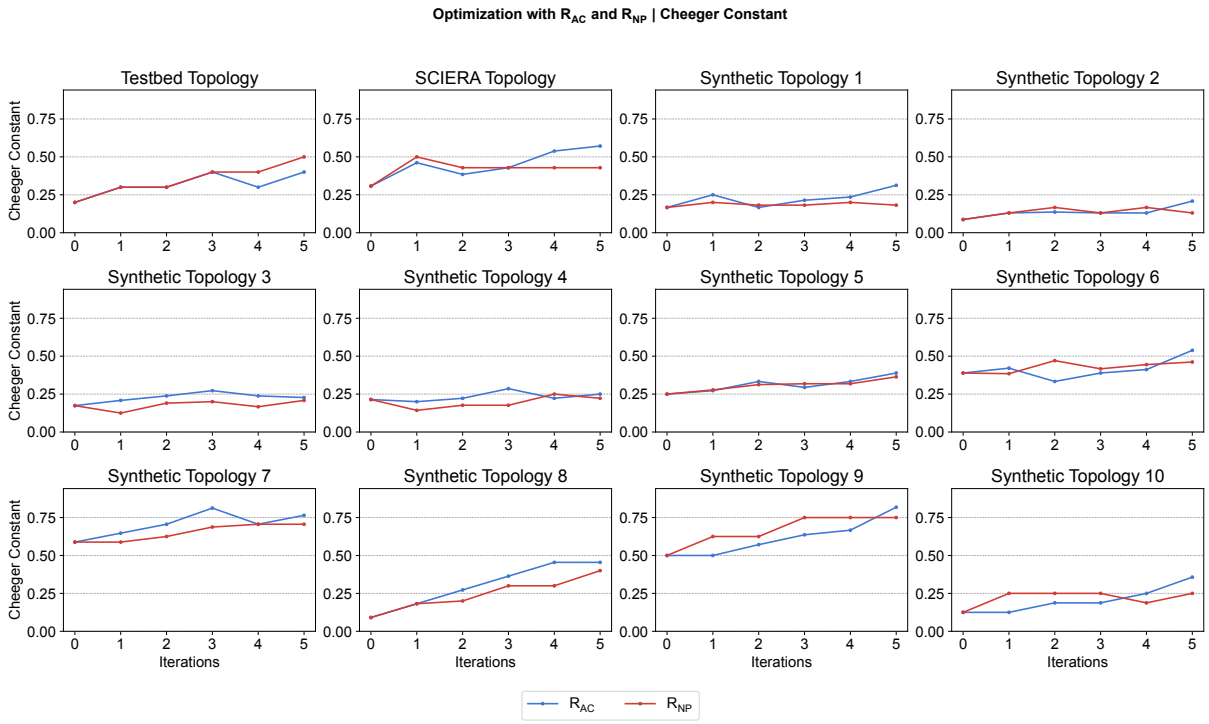

Figure 7.5: Cheeger constant over iterations of optimization algorithms R_{AC} and R_{NP} for all topologies.

this evaluation, the Cheeger constant is derived using the FM algorithm described in Chapter 5. Therefore, these results are non-deterministic approximations.

R_{AC} and R_{NP} perform similarly with regard to the Cheeger constant. Both algorithms successfully increase the metric, while neither output monotonically increases, and the results are generally unstable. It is notable that, despite being very different procedures, the results of R_{AC} and R_{NP} correlate closely with each other. Further, this synchronized behavior is not mirrored by the algebraic connectivity in Figure 7.4.

Both with regard to algebraic connectivity and the Cheeger constant, R_{NP} produces poor results for the SCIERA topology. In the case of the Testbed Topology, R_{NP} outperforms R_{AC} in both cases.

Border Breadth

Figure 7.6 shows the Border Breadth of the ISDs in the Testbed Topology. While R_{AC} and R_{NP} make different choices, they perform similarly well, with R_{NP} performing slightly better.

Figure 7.7 shows Synthetic Topology 8 and Synthetic Topology 9. In the former case, R_{NP} outperforms R_{AC} on average. However, it is notable that R_{NP} produces a steep increase in the Border Breadth of ISD 3, while the increases of ISD 1 and 2 are more moderate. R_{AC} shows a more even increase across the ISDs.

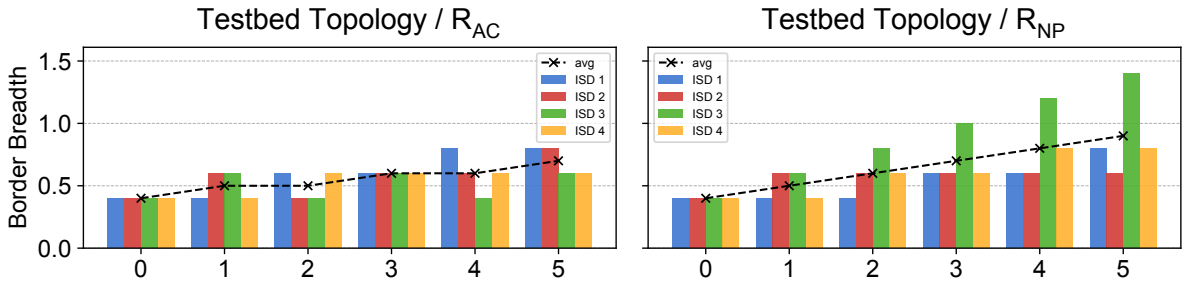

Figure 7.6: Border Breadth results on the Testbed Topology.

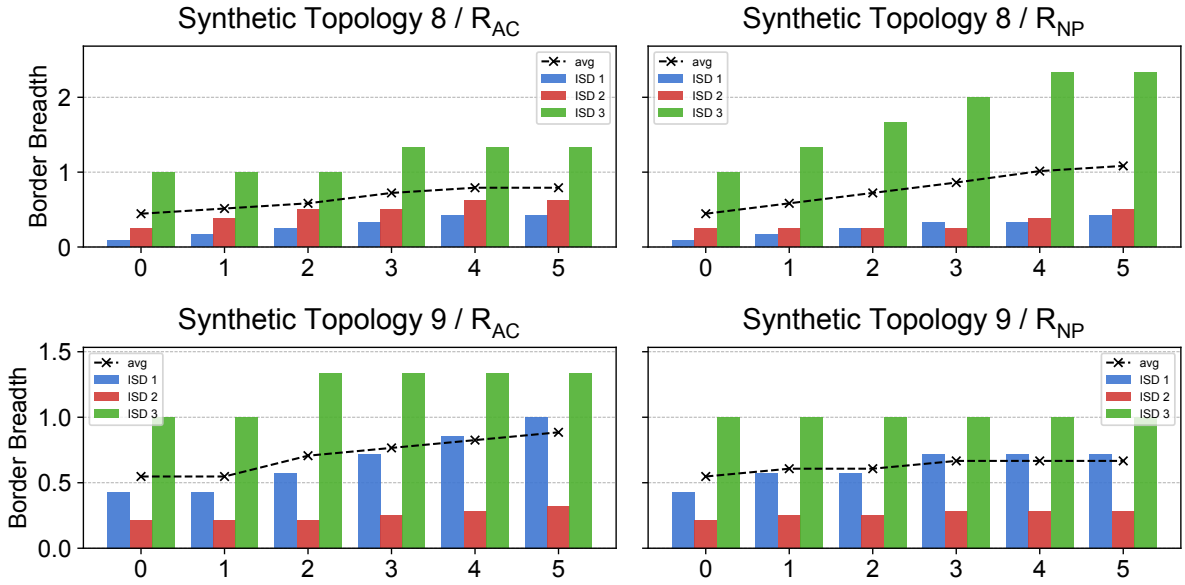

Figure 7.7: Border Breadth results for synthetic topologies 8 and 9.

Furthermore, Figure 7.7 shows that in the case of Synthetic Topology 9, R_{AC} performs well, while R_{NP} only produces a very slight increase. Results for all 12 baseline topologies are omitted here, but can be found in Figure A.1 and Figure A.2 in the appendix.

Figure 7.8 shows the average Border Breadth of all topologies over the course of the optimizations. Overall, both algorithms succeed in increasing the Border Breadth in most cases, though sometimes it stays constant. The only recorded decrease occurs in the last iteration of R_{AC} on Synthetic Topology 3. Notably, the overall positive impact on the Border Breadth is merely a side-effect of the algorithms, as both are agnostic to the metric.

7.2.2 Path Evaluation

This section details the experiments conducted by deploying both the baseline topologies and their altered versions on the testbed and collecting empirical measurements. Using the automated deployment procedure of the testbed, all 132 topologies were sequentially evaluated. For each deployment, the required nodes are built and then deployed. The

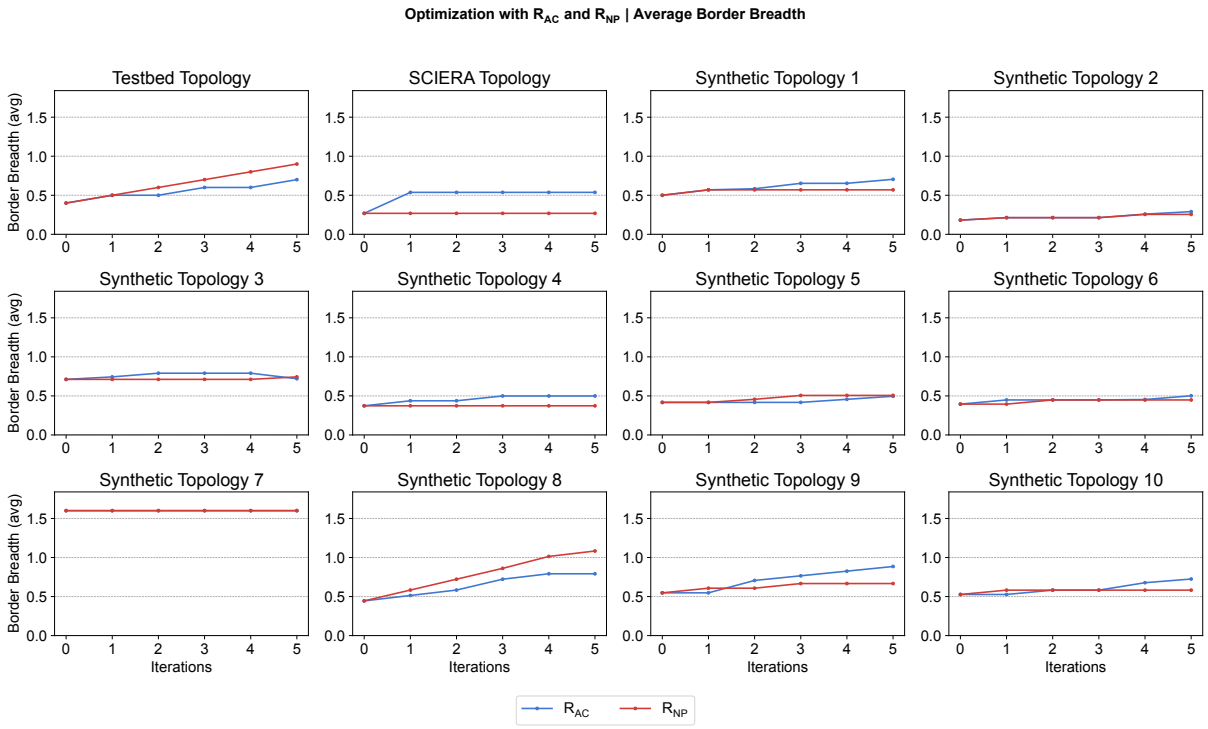

Figure 7.8: Average Border Breadth of all topologies during optimizations.

system waits for 30 seconds to ensure all connections have been established. Then, all SCION paths between each pair of nodes are evaluated. Finally, the average number of paths connecting a pair of nodes is calculated. For this analysis, a total of 179762 node pairs were evaluated. Of these, $\sim 0.74\%$ of evaluations failed.

Figure 7.9 shows the average path number over the course of the optimization processes for each baseline topology. As the figure shows, both algorithms have a positive effect on the average number of paths between nodes. The path average does not monotonically increase and is notably unstable in the case of Synthetic Topology 7. Furthermore, the average path count oscillates upwards for the initial Testbed Topology over the course of the optimization process by R_{AC}. This matches the curve of the Initial Testbed Topology under R_{AC} for both the algebraic connectivity and the Cheeger constant. Again, R_{NP} performs poorly on the SCIERA topology, resulting in a very slight reduction in the average number of paths.

In the case of the Synthetic Topology 3, R_{AC} produced a decrease in average path count after iteration 2, though the final state still constitutes a slight improvement. This matches the decrease in the Cheeger constant in Figure 7.5 and average Border Breadth in Figure 7.8. The algebraic connectivity in Figure 7.4 does not show the same decrease, which implies that a negative change in the topology was not captured by this metric.

Table 7.2 shows the overall gain in average path count for both optimization processes. Overall, it is apparent that the optimization produces a substantial improvement in path diversity for many topologies. Node pairs in the SCIERA Topology gain a total of 15.3 connecting paths on average after the optimization through R_{AC}. Between R_{AC} and R_{NP}, R_{AC} produces superior results. Notably, R_{AC} performs poorly in the case of Topology

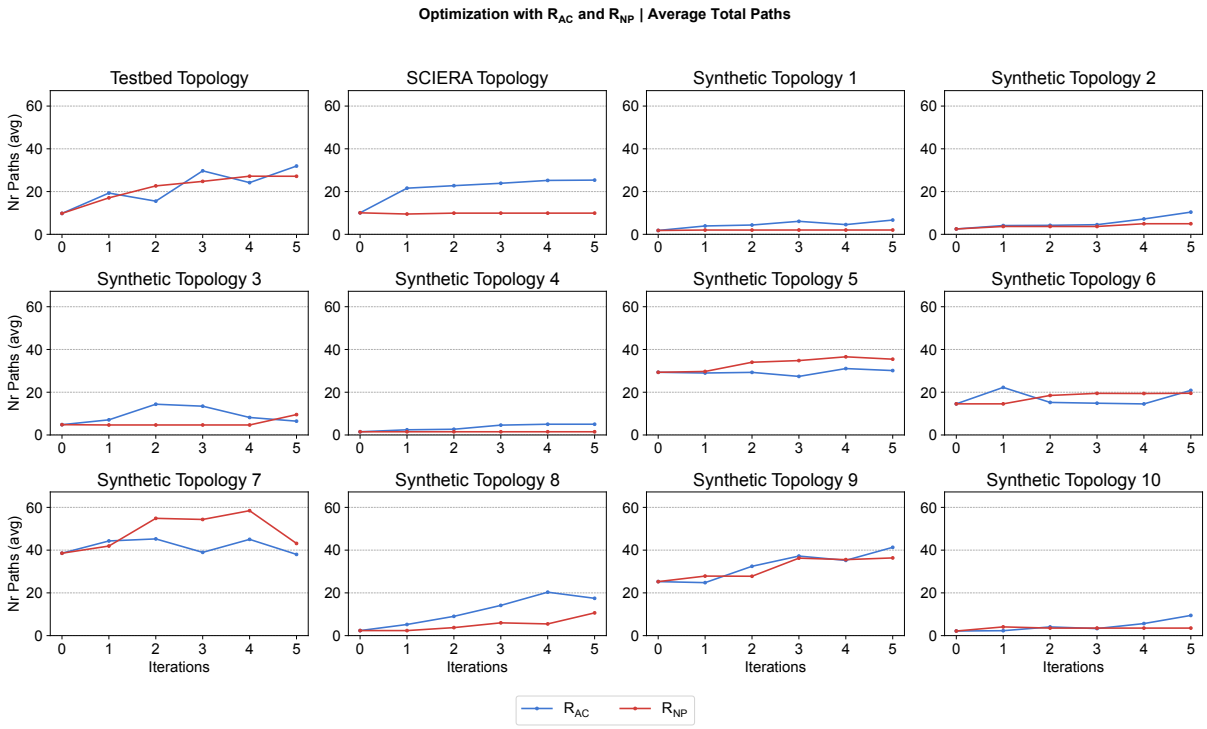

Figure 7.9: Average number of paths between all node pairs as topologies are optimized through R_{AC} and R_{NP}.

7, reducing the average path count by 0.55. This is not matched by the results shown in algebraic connectivity and the Cheeger constant. This implies that certain aspects of path optimizations are not captured by the metrics.

In a subsequent step, the node pairs were divided into inter-ISD and intra-ISD pairs. Figure 7.10 shows the average path count over the course of the evaluation for both inter-ISD and intra-ISD pairs. Results show that the curves for the two groups closely match. This contradicts the expectation that inter-ISD and intra-ISD paths exhibit a tradeoff. Rather, effective optimizations have a positive effect on both domains.

With the gains in average paths established, a further analysis examines the correlation between average path count and robustness metrics across all topologies. Figure 7.11 shows the corresponding results, with metrics sorted by correlation strength.

Overall, the Cheeger constant exhibits the strongest correlation with all types of path count. The algebraic connectivity shows the second strongest correlation. The average degree, which stays fixed over the course of the optimization, exhibits the third strongest correlation. Notably, algebraic connectivity and the Cheeger constant are more strongly correlated with the total average path count than the average degree.

The following section concludes this chapter by further discussing the results of the evaluation.

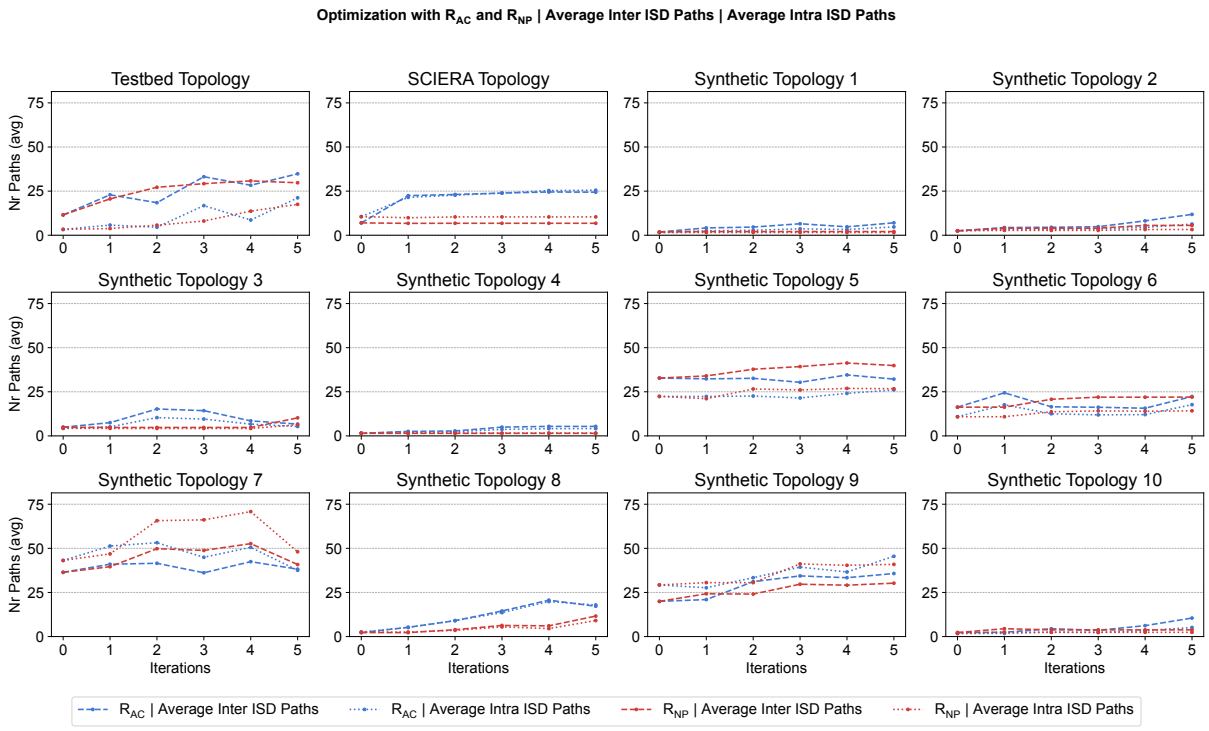

Figure 7.10: Average number of paths between all inter-ISD and intra-ISD node pairs as topologies are optimized through R_{AC} and R_{NP}.

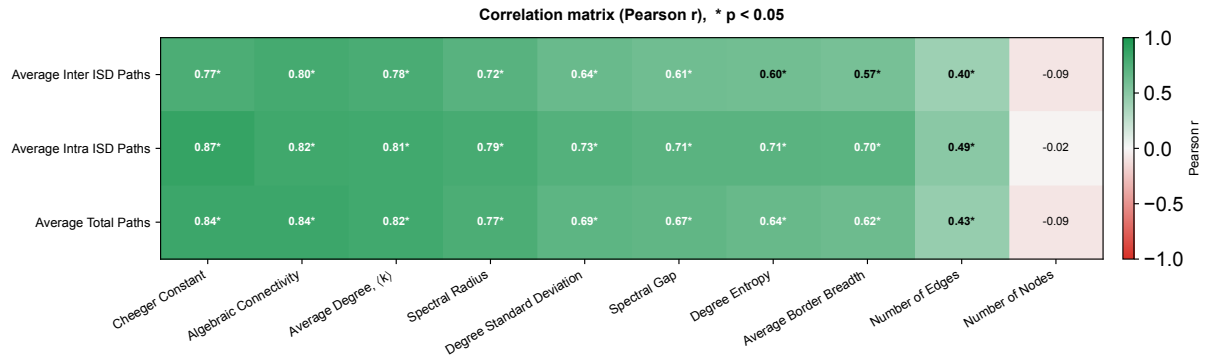

Figure 7.11: Correlations between graph metrics and average path counts, measured by Pearson r.

Topology	Initial Avg. Nr. Paths	Final Avg. Nr. Paths		Δ	
		R_{AC}	R_{NP}	R_{AC}	R_{NP}
Testbed Topology	9.79	31.94	27.16	+22.15	+17.37
SCIERA Topology	10.06	25.36	9.91	+15.30	-0.15
Synthetic Topology 1	1.81	6.68	2.01	+4.87	+0.21
Synthetic Topology 2	2.50	10.38	4.95	+7.88	+2.45
Synthetic Topology 3	4.82	6.47	9.53	+1.65	+4.72
Synthetic Topology 4	1.50	5.03	1.50	+3.53	+0.00
Synthetic Topology 5	29.33	30.13	35.46	+0.80	+6.12
Synthetic Topology 6	14.56	20.85	19.52	+6.30	+4.96
Synthetic Topology 7	38.55	38.00	43.13	-0.55	+4.58
Synthetic Topology 8	2.36	17.46	10.64	+15.10	+8.28
Synthetic Topology 9	25.27	41.31	36.35	+16.05	+11.09
Synthetic Topology 10	2.16	9.45	3.52	+7.28	+1.36

Table 7.2: Total gain in average paths between node pairs after running R_{AC} and R_{NP}.

7.3 Discussion

The experimental results show that both R_{AC} and R_{NP} successfully improve the path diversity in SCION networks. Overall, R_{AC} produced superior results, substantially increasing the average number of paths between node pairs with only five edge swaps. This confirmed the initial assumption established in Chapter 4 that a small number of edge rewirings can significantly improve censorship resilience. At the same time, results vary significantly across topologies.

It should also be noted that the examined topologies are relatively small, with the largest having $|V| = 49$. This makes it feasible to apply algorithms with higher time complexity than R_{AC} and R_{NP}, potentially yielding closer approximations or even optimal rewiring decisions. In the context of the overall evaluation process, which includes the deployment of the topologies and the logging of all paths between all node pairs, the computational costs of R_{AC} and R_{NP} are negligible.

Furthermore, the evolution of intra-ISD paths and inter-ISD paths closely match throughout the optimization process. This shows that there is no significant trade-off between inter-ISD and intra-ISD path diversity. This finding suggests that global optimizations can also benefit local domains. Consequently, in organizations involving multiple ISDs, such effects should be considered in decision-making processes.

While the average number of paths correlates closely with the Cheeger constant and the algebraic connectivity overall, some diverging results between the path evaluation and the robustness metrics were observed. This could imply that certain aspects of SCION path optimization are not captured by the metrics. The discrepancy could potentially arise from fine-grained SCION path policies that are not captured by the generalized robustness metrics.

Finally, it is important to note that the configuration of the baseline topologies has a

significant impact on the analysis and its results. Given the complexity of the evaluation pipeline, only a limited sample size could be considered. In a purely theoretical analysis, a larger set of topologies could be evaluated. Further, the generation process for the synthesized networks may impact the results in ways that are difficult to predict.

7.3.1 Reflections on R_{NP}

While the effectiveness of R_{AC} and R_{NP} differs, their performance appears to be correlated overall. However, in the case of SCIERA, R_{AC} produces a substantial improvement, while R_{NP} performed very poorly. The precise reason for this discrepancy is unclear. Based on the average degrees of the baseline topologies listed in Table 7.1, it is plausible that R_{NP} performs poorly on very sparse networks. This constitutes a major drawback in terms of real-world applicability.

In the experimental process, it was further observed that R_{NP} is sensitive to random factors. During repeated runs of the algorithm, the output may vary substantially. Further, it was observed that increasing the number of passes performed by the network partitioning has a less significant, though still favorable, impact on the output. This implies that the network partitioning finds good minimum cut approximations in few passes. While the minimum cut is reliable, not all minimum cuts are equally desirable, as discussed in Chapter 5. Therefore, the heuristics in the overarching logic of the divide and conquer strategies could still be improved. Furthermore, the algorithm might benefit from an initialization strategy, where the currently random initial partitioning of the network is replaced with a strategic choice.

7.3.2 Correlations

When considering the correlation results shown in Figure 7.11, it must be noted that they are derived from a sample size of $N = 132$, where the 132 graphs are structurally similar. Overall, the 12 baseline topologies are quite similar in terms of average degree, edge count, and size, as shown in Table 7.1. Furthermore, the average degree and the number of nodes and edges stay consistent with the baseline topologies during the optimization process. Additionally, 10 baseline topologies were generated through the same process.

The improvements achieved through rewiring are further supported by the observation that the Cheeger constant and the algebraic connectivity exhibit a stronger correlation to average path count than the density of the networks. While the difference is small, this implies that a well-optimized network can have greater censorship resilience than a graph of higher density, confirming the conclusion from Chapter 4.

The correlation matrix further suggests that algebraic connectivity and the Cheeger constant were effective guiding metrics for the optimization. However, as the algorithms were designed to optimize these metrics, the correlations produced are limited to this context and do not necessarily show that they are the best suited robustness metrics to predict path redundancy in every scenario. While the two metrics' strong correlation with path

counts may in part be due to them being highly indicative of high path redundancy, it may also show that the algorithms very effectively improve the algebraic connectivity and the Cheeger constant above other metrics.

7.3.3 Border Breadth

The average Border Breadth similarly bears a positive correlation to the path counts; however, it is weaker compared to the algebraic connectivity and the Cheeger constant. Further, among the path classes, it exhibits the strongest correlation with inter-ISD routing paths, which aligns with expectations. Somewhat less expected is the nevertheless positive correlation with intra-ISD routing paths, considering that there exists a trade-off between Border Breath and intra-ISD edge count. However, this correlation is likely a secondary effect, as R_{NP} and R_{AC} produce an increase in average path counts overall, while often increasing Border Breadth.

It should also be noted that the weaker correlation between average path counts and Border Breadth may be related to the properties of the evaluated graphs. For example, ISDs are likely to remain at a small Border Breadth if they only have one core node. At the same time, the synthetic topologies may already have close to optimal Border Breadths as a mere consequence of the generation process. An evaluation of the impact of the Border Breadth may therefore be more salient in a context where the set of core nodes can be modified by the optimization process, or when less favorable initial states are examined.

Chapter 8

Conclusion

This thesis is concerned with the censorship resilience potential of the SCION architecture. SCION partitions ASes into ISDs, which may increase censorship vulnerability on the AS-routing level. Conversely, SCION's multipath capabilities can enhance censorship resilience compared to BGP. These topological factors were therefore investigated in detail through both analytical and empirical methods. To date, empirical research on the censorship resilience of FIAs remains scarce; this thesis contributes to addressing this gap through a detailed empirical analysis.

The contributions of this work are as follows: (1) A survey of robustness metrics and a novel taxonomy; (2) A theoretical robustness analysis of the BGP network, a hypothetical SCION topology, and expander graphs; (3) the introduction of the novel metric Border Breadth; (4) a SCION-specific adaptation of an existing algorithm, resulting in the optimization algorithm R_{AC}; (5) a novel optimization algorithm R_{NP}; (6) a simulation testbed enabling the easy deployment of user-configured SCION topologies, including tools for experimentation; and (7) practical experiments with R_{AC} and R_{NP} on the simulation testbed.

For the theoretical analysis, the real-world BGP network was compared to a hypothetical SCION topology using various robustness metrics. Expander graphs served as an optimal benchmark. In this evaluation, expander graphs outperformed all other networks by a wide margin, despite being significantly sparser than some of the compared topologies. These results indicate that topological modifications can substantially enhance censorship resilience while keeping the number of edges in Internet topologies constant.

For the experimental evaluation, a set of baseline topologies underwent optimization using both R_{AC} and R_{NP}. The resulting topologies were analyzed using robustness metrics and deployed on the testbed. SCION paths connecting nodes were empirically measured. Overall, the results show that both algorithms increase the average number of paths between nodes, with R_{AC} outperforming R_{NP}. Performance varied across the 24 final topologies, many exhibiting substantial improvements, and with slight decreases being produced in two instances.

By applying the optimization process R_{AC} to the topology of the real-world SCION network SCIERA, a concrete rewired topology was obtained that increases the average number of paths between node pairs by 15.3 after five edge swaps.

Further, it was found that while the novel algorithm R_{NP} produces favorable results in some cases and may sometimes outperform R_{AC}, it also underperforms in many other instances. This suggests that it is better suited to certain topologies, although the decisive structural properties have not yet been identified. Additionally, R_{NP} is sensitive to its random initialization.

The *divide-and-conquer* approach using a fast network partitioning approximation remains an interesting research direction, as it could potentially yield precise approximations in $O(|V| \log |V|)$ time, if properly optimized. However, a key challenge arises: while partitioning approximations like the Fiduccia-Mattheysis algorithm effectively produce minium cut approximations, not all minimum partitions are equally desirable when operating on subgraphs. This is why the tie-breaking strategy used by R_{NP} is essential. Nevertheless, improved heuristics and selection procedures may be developed in future work.

Over the course of this work, a central trade-off between SCION design goals and path redundancy was identified, which reflects a fundamental tension in network design: networks with high expansion and path redundancy tend to be more susceptible to certain security risks, such as failure propagation. Balancing these competing objectives might constitute a key challenge for censorship resilience in FIAs. More broadly, this mirrors the well-established trade-off between privacy and security.

8.1 Limitations

While the theoretical analysis used real-world BGP topologies, the SCION topology was derived from the same dataset and only represents a hypothetical topology. Furthermore, in order to apply complex robustness metrics, the BGP topology was downsampled, which may introduce biases.

The testbed environment enables easily configurable topologies to be deployed. It further includes tools for automated, sequential experiments on a set of topologies. The Docker containers used to simulate ASes require little memory, allowing for the deployment of large test networks. However, during sequential experiments, the time required for repeated building and deployment of containers introduce a bottleneck. These issues could be addressed by optimizing the deployment process, or by enabling topology modifications on a running network, thereby eliminating the need for repeated deployments.

Furthermore, the optimization processes used in this work are limited to edge rewirings. The configuration of ISDs and core ASes is assumed to be fixed. By strategically restructuring ISDs and designating core ASes, significant improvements in path redundancy could be achieved. However, due to the aforementioned trade-off between the SCION design goals and maximal path redundancy, a network composed solely of core nodes might achieve optimal path redundancy while undermining the purpose of the architecture. For

similar reasons, multi-ISD ASes and peer-links across ISDs were intentionally excluded from this examination. This, however, constitutes a further limitation.

Finally, the results of experiments in Chapter 7 were produced using a set of 12 small networks, with $|V| < 50$. The networks were relatively similar in average degree, and 10 of them were synthesized by the same generation procedure, resulting in potential biases.

8.2 Future Work

Future research could conduct similar experiments with a wider range of topologies. Further, it could also explore optimization techniques more closely tailored to SCION, for example, by considering the semantic role of core nodes, ISD segmentation, and the hierarchical structure of communication paths. Additionally, applying a broader array of existing optimization techniques may yield improved results. More complex algorithms are also feasible in this context and could yield more optimal solutions.

From an algorithmic perspective, further research on combining network partitioning approximations with a *divide-and-conquer* approach may be valuable. Importantly, this problem extends beyond SCION and censorship resilience, and is more generally rooted in graph theory, making it potentially applicable to a wide range of domains.

From a technical perspective, adapting the current testbed solution to support topological changes during runtime would constitute a valuable evolution of the testing environment.

More broadly, research on censorship resilience in FIAs remains limited. Further work addressing the aforementioned trade-off between failure propagation and censorship resilience would be valuable. Additionally, a systematic comparison and further empirical studies of different FIAs with respect to censorship resilience remain an open problem.

Bibliography

[1] K. Vesteinsson and G. Baker, *An uncertain future for the global internet*, 2025. [Online]. Available: `https://freedomhouse.org/report/freedom-net/2025/uncertain-future-global-internet`.

[2] L. Salamatian, F. Douzet, K. Salamatian, and K. Limonier, "The geopolitics behind the routes data travel: A case study of iran," *Journal of Cybersecurity*, vol. 7, no. 1, tyab018, 2021.

[3] R. Singh, H. Koo, N. Miramirkhani, F. Mirhaj, P. Gill, and L. Akoglu, "The politics of routing: Investigating the relationship between {as} connectivity and internet freedom," in *6th USENIX Workshop on Free and Open Communications on the Internet (FOCI 16)*, 2016.

[4] K. G. Leyba, B. Edwards, C. Freeman, J. R. Crandall, and S. Forrest, "Borders and gateways: Measuring and analyzing national as chokepoints," in *Proceedings of the 2nd ACM SIGCAS Conference on Computing and Sustainable Societies*, 2019, pp. 184–194.

[5] X. Xu, Z. M. Mao, and J. A. Halderman, "Internet censorship in china: Where does the filtering occur?" In *International Conference on Passive and Active Network Measurement*, Springer, 2011, pp. 133–142.

[6] M. Wrana, D. Barradas, and N. Asokan, "Sok: The spectre of surveillance and censorship in future internet architectures," *Proceedings on Privacy Enhancing Technologies*, 2025.

[7] M. Ivanović, F. Wirz, J. S. Nieto, and A. Perrig, "Charting censorship resilience and global internet reachability: A quantitative approach," in *2024 IFIP Networking Conference (IFIP Networking)*, IEEE, 2024, pp. 529–535.

[8] F. Wirz et al., "Scaling sciera: A journey through the deployment of a next-generation network," in *Proceedings of the ACM SIGCOMM 2025 Conference*, 2025, pp. 720–741.

[9] S. Schuster, M. Van Den Berg, X. Larrucea, T. Slewe, and P. Ide-Kostic, "Mass surveillance and technological policy options: Improving security of private communications," *Computer Standards & Interfaces*, vol. 50, pp. 76–82, 2017.

[10] V. Ververis, L. Lasota, T. Ermakova, and B. Fabian, "Website blocking in the european union: Network interference from the perspective of open internet," *Policy & Internet*, vol. 16, no. 1, pp. 121–148, 2024.

[11] A. Master and C. Garman, "A worldwide view of nation-state internet censorship," *Free and Open Communications on the Internet*, 2023.

[12] C. S. Leberknight, M. Chiang, H. V. Poor, and F. Wong, "A taxonomy of internet censorship and anti-censorship."

[13] S. A. Meserve and D. Pemstein, "Google politics: The political determinants of internet censorship in democracies," *Political Science Research and Methods*, vol. 6, no. 2, pp. 245–263, 2018.

[14] S. Aryan, H. Aryan, and J. A. Halderman, "Internet censorship in iran: A first look," in *3rd USENIX Workshop on Free and Open Communications on the Internet (FOCI 13)*, 2013.

[15] T. Anderson et al., "A brief overview of the nebula future internet architecture," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 3, pp. 81–86, 2014.

[16] L. Zhang et al., "Named data networking," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 3, pp. 66–73, 2014.

[17] D. Raychaudhuri, K. Nagaraja, and A. Venkataramani, "Mobilityfirst: A robust and trustworthy mobility-centric architecture for the future internet," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 16, no. 3, pp. 2–13, 2012.

[18] A. Anand et al., "Xia: An architecture for an evolvable and trustworthy internet," in *Proceedings of the 10th ACM Workshop on Hot Topics in Networks*, 2011, pp. 1–6.

[19] D. Barrera, L. Chuat, A. Perrig, R. M. Reischuk, and P. Szalachowski, "The scion internet architecture," *Communications of the ACM*, vol. 60, no. 6, pp. 56–65, 2017.

[20] Y. Rekhter, T. Li, and S. Hares, "A border gateway protocol 4 (bgp-4)," Tech. Rep., 2006.

[21] SCIERA Project, *Sciera documentation*, `https://sciera.readthedocs.io/en/latest/`, Accessed: March 27, 2026.

[22] M. Ambrosin, A. Compagno, M. Conti, C. Ghali, and G. Tsudik, "Security and privacy analysis of nsf future internet architectures," *arXiv preprint arXiv:1610.00355*, 2016.

[23] W. Ding, Z. Yan, and R. H. Deng, "A survey on future internet security architectures," *IEEE Access*, vol. 4, pp. 4374–4393, 2016.

[24] C. Chen, D. E. Asoni, D. Barrera, G. Danezis, and A. Perrig, "Hornet: High-speed onion routing at the network layer," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 2015, pp. 1441–1454.

[25] H. B. Acharya, S. Chakravarty, and D. Gosain, "Few throats to choke: On the current structure of the internet," in *2017 IEEE 42nd conference on local computer networks (LCN)*, IEEE, 2017, pp. 339–346.

[26] H. Roberts, D. Larochelle, R. Faris, and J. Palfrey, "Mapping local internet control," in *Computer Communications Workshop (Hyannis, CA, 2011), IEEE*, 2011.

[27] D. Gosain, A. Agarwal, S. Chakravarty, and H. B. Acharya, "The devil's in the details: Placing decoy routers in the internet," in *Proceedings of the 33rd Annual Computer Security Applications Conference*, 2017, pp. 577–589.

[28] R. S. Raman, L. Evdokimov, E. Wurstrow, J. A. Halderman, and R. Ensafi, "Investigating large scale https interception in kazakhstan," in *Proceedings of the ACM Internet Measurement Conference*, 2020, pp. 125–132.

[29] J. Levett, V. Vassilakis, and P. Yadav, "Unveiling internet censorship: Analysing the impact of nation states' content control efforts on internet architecture and routing patterns," *arXiv preprint arXiv:2402.19375*, 2024.

[30] X. Liang, G. Liu, L. Jin, S. Hao, and H. Wang, "Pathfinder: Exploring path diversity for assessing internet censorship inconsistency," *arXiv preprint arXiv:2407.04213*, 2024.

[31] A. Sydney, C. Scoglio, and D. Gruenbacher, "Optimizing algebraic connectivity by edge rewiring," *Applied Mathematics and computation*, vol. 219, no. 10, pp. 5465–5479, 2013.

[32] D. Mosk-Aoyama, "Maximum algebraic connectivity augmentation is np-hard," *Operations Research Letters*, vol. 36, no. 6, pp. 677–679, 2008.

[33] R. Albert, H. Jeong, and A.-L. Barabási, "Error and attack tolerance of complex networks," *nature*, vol. 406, no. 6794, pp. 378–382, 2000.

[34] C. M. Schneider, A. A. Moreira, J. S. Andrade Jr, S. Havlin, and H. J. Herrmann, "Mitigation of malicious attacks on networks," *Proceedings of the National Academy of Sciences*, vol. 108, no. 10, pp. 3838–3841, 2011.

[35] V. H. Louzada, F. Daolio, H. J. Herrmann, and M. Tomassini, "Smart rewiring for network robustness," *Journal of Complex networks*, vol. 1, no. 2, pp. 150–159, 2013.

[36] V. H. Louzada, F. Daolio, H. J. Herrmann, and M. Tomassini, "Generating robust and efficient networks under targeted attacks," in *Propagation Phenomena in Real World Networks*, Springer, 2015, pp. 215–224.

[37] S. Kumari, A. Saroha, and A. Singh, "Efficient edge rewiring strategies for enhancement in network capacity," *Physica A: Statistical Mechanics and its Applications*, vol. 545, p. 123 552, 2020.

[38] L. Rong and J. Liu, "A heuristic algorithm for enhancing the robustness of scale-free networks based on edge classification," *Physica A: Statistical Mechanics and its Applications*, vol. 503, pp. 503–515, 2018.

[39] S. Kumari, R. Kumar, S. Muhuri, and S. Namasudra, "A novel budget-constrained rewiring strategy for information flow in social networks," *IEEE Transactions on Computational Social Systems*, vol. 12, no. 5, pp. 2242–2253, 2024.

[40] M. Zhou and J. Liu, "A memetic algorithm for enhancing the robustness of scale-free networks against malicious attacks," *Physica A: Statistical Mechanics and its Applications*, vol. 410, pp. 131–143, 2014.

[41] M. Chujyo and Y. Hayashi, "A loop enhancement strategy for network robustness," *Applied Network Science*, vol. 6, no. 1, p. 3, 2021.

[42] H. Chan and L. Akoglu, "Optimizing network robustness by edge rewiring: A general framework," *Data Mining and Knowledge Discovery*, vol. 30, no. 5, pp. 1395–1425, 2016.

[43] M. Oehlers and B. Fabian, "Graph metrics for network robustness—a survey," *Mathematics*, vol. 9, no. 8, p. 895, 2021.

[44] S. Freitas, D. Yang, S. Kumar, H. Tong, and D. H. Chau, "Graph vulnerability and robustness: A survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 6, pp. 5915–5934, 2022.

[45] Y. Lou, L. Wang, and G. Chen, "Structural robustness of complex networks: A survey of a posteriori measures [feature]," *IEEE Circuits and Systems Magazine*, vol. 23, no. 1, pp. 12–35, 2023.

[46] B. Wang, H. Tang, C. Guo, and Z. Xiu, "Entropy optimization of scale-free networks' robustness to random failures," *Physica A: Statistical Mechanics and its Applications*, vol. 363, no. 2, pp. 591–596, 2006.

[47] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relationships of the internet topology," *ACM SIGCOMM computer communication review*, vol. 29, no. 4, pp. 251–262, 1999.

[48] S.-T. Park, D. M. Pennock, and C. L. Giles, "Comparing static and dynamic measurements and models of the internet's as topology," in *IEEE INFOCOM 2004*, IEEE, vol. 3, 2004, pp. 1616–1627.

[49] R. Criado, J. Flores, B. Hernández-Bermejo, J. Pello, and M. Romance, "Effective measurement of network vulnerability under random and intentional attacks," *Journal of Mathematical Modelling and Algorithms*, vol. 4, no. 3, pp. 307–316, 2005.

[50] B. Mburano, W. Si, and W. X. Zheng, "A comparative study on the variants of r metric for network robustness," in *2021 International Symposium on Networks, Computers and Communications (ISNCC)*, IEEE, 2021, pp. 1–6.

[51] M. J. Alenazi, E. K. Cetinkaya, and J. P. Sterbenz, "Cost-constrained and centrality-balanced network design improvement," in *2014 6th International Workshop on Reliable Networks Design and Modeling (RNDM)*, IEEE, 2014, pp. 194–201.

[52] M. J. Alenazi and J. P. Sterbenz, "Comprehensive comparison and accuracy of graph metrics in predicting network resilience," in *2015 11th international conference on the design of reliable communication networks (DRCN)*, IEEE, 2015, pp. 157–164.

[53] M. E. Newman, "Scientific collaboration networks. i. network construction and fundamental results," *Physical review E*, vol. 64, no. 1, p. 016 131, 2001.

[54] M. E. Newman, "Assortative mixing in networks," *Physical review letters*, vol. 89, no. 20, p. 208 701, 2002.

[55] P. Mahadevan, D. Krioukov, M. Fomenkov, X. Dimitropoulos, K. Claffy, and A. Vahdat, "The internet as-level topology: Three data sources and one definitive metric," *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 1, pp. 17–26, 2006.

[56] W. Ellens and R. E. Kooij, "Graph measures and network robustness," *arXiv preprint arXiv:1311.5064*, 2013.

[57] S. N. Soffer and A. Vazquez, "Network clustering coefficient without degree-correlation biases," *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, vol. 71, no. 5, p. 057 101, 2005.

[58] M. E. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical review E*, vol. 69, no. 2, p. 026 113, 2004.

[59] C. Gkantsidis, M. Mihail, and E. Zegura, "Spectral analysis of internet topologies," in *IEEE INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Cat. No. 03CH37428)*, IEEE, vol. 1, 2003, pp. 364–374.

[60] A. H. Dekker, B. D. Colbert, et al., "Network robustness and graph topology," in *ACSC*, vol. 4, 2004, pp. 359–368.

[61] F. Harary, *Graph Theory*. Reading, MA: Addison-Wesley, 1969.

[62] L. W. Beineke, O. R. Oellermann, and R. E. Pippert, "The average connectivity of a graph," *Discrete mathematics*, vol. 252, no. 1-3, pp. 31–45, 2002.

[63] R. Diestel, *Graph theory*. Springer Nature, 2025.

[64] H. Frank and I. T. Frisch, "Network analysis," *Scientific American*, vol. 223, no. 1, pp. 94–105, 1970.

[65] F. Harary, "Conditional connectivity," *Networks*, vol. 13, no. 3, pp. 347–357, 1983.

[66] S. Sreenivasan, R. Cohen, E. López, Z. Toroczkai, and H. E. Stanley, "Structural bottlenecks for communication in networks," *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, vol. 75, no. 3, p. 036 105, 2007.

[67] J. Cheeger, "A lower bound for the smallest eigenvalue of the laplacian," in *Problems in Analysis: A Symposium in Honor of Salomon Bochner (PMS-31)*, Princeton University Press, 1971, pp. 195–200, ISBN: 978-1-4008-6931-2. DOI: 10.1515/9781400869312-013.

[68] C. M. Fiduccia and R. M. Mattheyses, "A linear-time heuristic for improving network partitions," in *Papers on Twenty-Five Years of Electronic Design Automation*, ACM, 1988, pp. 241–247.

[69] D. Moazzami and S. Salehian, "On the edge-tenacity of graphs," in *Int. Math. Forum*, vol. 3, 2008, pp. 929–936.

[70] R. M. Salles and D. A. Marino, "Strategies and metric for resilience in computer networks," *The Computer Journal*, vol. 55, no. 6, pp. 728–739, 2012.

[71] M. J. Lipman and R. Pippert, "Toward a measure of vulnerability ii. the ratio of disruption," in *Graph theory with applications to algorithms and computer science*, 1985, pp. 507–517.

[72] W. Si, B. Mburano, W. X. Zheng, and T. Qiu, "Measuring network robustness by average network flow," *IEEE Transactions on Network Science and Engineering*, vol. 9, no. 3, pp. 1697–1712, 2022.

[73] J. S. Baras and P. Hovareshti, "Efficient and robust communication topologies for distributed decision making in networked systems," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, IEEE, 2009, pp. 3751–3756.

[74] F. De Montgolfier, M. Soto, and L. Viennot, "Treewidth and hyperbolicity of the internet," in *2011 IEEE 10th International Symposium on Network Computing and Applications*, IEEE, 2011, pp. 25–32.

[75] J. P. Rohrer, A. Jabbar, and J. P. Sterbenz, "Path diversification: A multipath resilience mechanism," in *2009 7th International Workshop on Design of Reliable Communication Networks*, IEEE, 2009, pp. 343–351.

[76] H. Tong, B. A. Prakash, C. Tsourakakis, T. Eliassi-Rad, C. Faloutsos, and D. H. Chau, "On the vulnerability of large graphs," in *2010 IEEE international conference on data mining*, IEEE, 2010, pp. 1091–1096.

[77] E. Estrada, "Network robustness to targeted attacks. the interplay of expansibility and degree distribution," *The European Physical Journal B-Condensed Matter and Complex Systems*, vol. 52, no. 4, pp. 563–574, 2006.

[78] C. R. Palmer, G. Siganos, M. Faloutsos, C. Faloutsos, and P. B. Gibbons, "The connectivity and fault-tolerance of the internet topology," *In Proceedings of the Workshop on Network Related Data Management (NRDM 2001)*, 2001.

[79] F. D. Malliaros, V. Megalooikonomou, and C. Faloutsos, "Fast robustness estimation in large social graphs: Communities and anomaly detection," in *Proceedings of the 2012 SIAM International Conference on Data Mining*, SIAM, 2012, pp. 942–953.

[80] E. F. Moore and C. E. Shannon, "Reliable circuits using less reliable relays," *Journal of the Franklin Institute*, vol. 262, no. 3, pp. 191–208, 1956.

[81] S. C. Liew and K. W. Lu, "A framework for network survivability characterization," in *[Conference Record] SUPERCOMM/ICC'92 Discovering a New World of Communications*, IEEE, 1992, pp. 405–410.

[82] Y. Sato, S. Ata, and I. Oka, "A strategic approach for re-organizing the internet topology by applying social behavior dynamics," *Journal of Network and Systems Management*, vol. 17, no. 1, pp. 208–229, 2009.

[83] Y. Lu, Y. Zhao, F. Sun, and R. Liang, "Measuring and improving communication robustness of networks," *IEEE Communications Letters*, vol. 23, no. 12, pp. 2168–2171, 2019.

[84] P. Crucitti, V. Latora, M. Marchiori, and A. Rapisarda, "Efficiency of scale-free networks: Error and attack tolerance," *Physica A: Statistical Mechanics and its Applications*, vol. 320, pp. 622–642, 2003.

[85] J. Liu, M. Zhou, S. Wang, and P. Liu, "A comparative study of network robustness measures," *Frontiers of Computer Science*, vol. 11, no. 4, pp. 568–584, 2017.

[86] B. Mohar, "Isoperimetric numbers of graphs," *Journal of combinatorial theory, Series B*, vol. 47, no. 3, pp. 274–291, 1989.

[87] P. Buser, "Cubic graphs and the first eigenvalue of a riemann surface," *Mathematische Zeitschrift*, vol. 162, no. 1, pp. 87–99, 1978.

[88] S. Hoory, N. Linial, and A. Wigderson, "Expander graphs and their applications," *Bulletin of the American Mathematical Society*, vol. 43, no. 4, pp. 439–561, 2006.

[89] F. Boesch and R. Thomas, "On graphs of invulnerable communication nets," *IEEE Transactions on Circuit Theory*, vol. 17, no. 2, pp. 183–192, 1970. DOI: 10.1109/TCT.1970.1083099.

[90] M. J. Alenazi and J. P. Sterbenz, "Evaluation and comparison of several graph robustness metrics to improve network resilience," in *2015 7th International Workshop on Reliable Networks Design and Modeling (RNDM)*, IEEE, 2015, pp. 7–13.

[91] W. Jun, M. Barahona, T. Yue-Jin, and D. Hong-Zhong, "Natural connectivity of complex networks," *Chinese physics letters*, vol. 27, no. 7, p. 078 902, 2010.

[92] D. Fay et al., "Weighted spectral distribution for internet topology analysis: Theory and applications," *IEEE/ACM Transactions on networking*, vol. 18, no. 1, pp. 164–176, 2009.

[93] X. Long, D. Tipper, and T. Gomes, "Measuring the survivability of networks to geographic correlated failures," *Optical Switching and Networking*, vol. 14, pp. 117–133, 2014.

[94] M. Fiedler, "Algebraic connectivity of graphs," *Czechoslovak mathematical journal*, vol. 23, no. 2, pp. 298–305, 1973.

[95] W. Ellens, F. M. Spieksma, P. Van Mieghem, A. Jamakovic, and R. E. Kooij, "Effective graph resistance," *Linear algebra and its applications*, vol. 435, no. 10, pp. 2491–2506, 2011.

[96] A. Tizghadam and A. Leon-Garcia, "On robust traffic engineering in core networks," in *IEEE GLOBECOM 2008-2008 IEEE Global Telecommunications Conference*, IEEE, 2008, pp. 1–6.

[97] H. Tangmunarunkit, R. Govindan, S. Jamin, S. Shenker, and W. Willinger, "Network topology generators: Degree-based vs. structural," *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 4, pp. 147–159, 2002.

[98] E. Estrada, "Spectral scaling and good expansion properties in complex networks," *Europhysics Letters*, vol. 73, no. 4, p. 649, 2006.

[99] A. Zeng and W. Liu, "Enhancing network robustness against malicious attacks," *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, vol. 85, no. 6, p. 066 130, 2012.

[100] A.-L. Barabási, "Chapter 8: Percolation and Network Robustness," in *Network Science*. 2012, November 2012 version.

[101] L. Spitz, *Theoretical internet topology analysis*, https://github.com/ringdinglinn/scion-analysis, Accessed: 2026-04-17, 2026.

[102] *The caida as relationships dataset*, https://www.caida.org/catalog/datasets/as-relationships/, Dataset 2025-12-01, CAIDA, Dec. 2025.

[103] *The caida as organizations dataset*, https://www.caida.org/catalog/datasets/as-organizations, CAIDA, Dec. 2025. DOI: 10.21986/CAIDA.DATA.AS-TO-ORG-MAPPING.

[104] *Caida as rank*, http://as-rank.caida.org/, CAIDA, Dec. 2025.

[105] D. Walton, A. Retana, E. Chen, and J. Scudder, "Advertisement of multiple paths in bgp," Tech. Rep., 2016.

[106] Y. Cui, X. Li, J. Li, H. Wang, and X. Chen, "A survey of sampling method for social media embeddedness relationship," *ACM Computing Surveys*, vol. 55, no. 4, pp. 1–39, 2022.

[107] V. Krishnamurthy, M. Faloutsos, M. Chrobak, J.-H. Cui, L. Lao, and A. G. Percus, "Sampling large internet topologies for simulation purposes," *Computer Networks*, vol. 51, no. 15, pp. 4284–4302, 2007.

[108] A. A. Hagberg, D. A. Schult, and P. J. Swart, "Exploring network structure, dynamics, and function using networkx," in *Proceedings of the 7th Python in Science Conference*, G. Varoquaux, T. Vaught, and J. Millman, Eds., Pasadena, CA USA, 2008, pp. 11–15.

[109] X. Zhang, H.-C. Hsiao, G. Hasker, H. Chan, A. Perrig, and D. G. Andersen, "Scion: Scalability, control, and isolation on next-generation networks," in *2011 IEEE Symposium on Security and Privacy*, IEEE, 2011, pp. 212–227.

[110] L. Spitz, *Scion topology testbed with dynamic topology configurability*, `https://github.com/ringdinglinn/scion-topology`, Accessed: 2026-04-17, 2026.

[111] C. Lanczos, "An iteration method for the solution of the eigenvalue problem of linear differential and integral operators," *Journal of research of the National Bureau of Standards*, vol. 45, no. 4, pp. 255–282, 1950.

[112] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU press, 2013.

[113] N. Isaak, N. Matumona, and K. Schlup, "Design and implementation of a SCION testbed for internet censorship research," Communication Systems Group, Department of Informatics, Universität Zürich, Zürich, Switzerland, Tech. Rep., Jan. 2026. [Online]. Available: `https://files.ifi.uzh.ch/CSG/staff/gruebl/extern/theses/map-isaak-schlup-matumona.pdf`.

[114] SCION Association, *Scion v0.14.0 release*, `https://github.com/scionproto/scion/releases/tag/v0.14.0`, Accessed: 2026-04-17, Nov. 2025.

Abbreviations

AS	Autonomous System
BA	Barabási-Albert
BFS-SM	Breadth-First-Search-Based Sampling Method
BGP	Border Gateway Protocol
CA	Certificate Authority
CAIDA	Center for Applied Internet Data Analysis
CKDBC	K-core (using Closeness centrality), Disassortativity and Betweenness Centrality
CLI	Command-Line Interface
CRP	Censorship Resilience Potential
CRVE	Contraction of Random Vertex/Edge
CS	Content Store
CSV	Comma-Separated Values
DEC	Disassortativity and Eigenvector Centrality
DKBC	K-core (using Degree) and Betweenness Centrality
DNS	Domain Name System
DoS	Denial-of-Service
DPA	Disassortativity and Preferential Attachment
FIA	Future Internet Architecture
FIB	Forwarding Information Base
FM	Fiduccia–Mattheyses
FM-R_{NP}	Fiduccia-Mattheyes in Rewire by Network Partition
FVS	Feedback Vertex Set
GNS	Global Name Service
GUID	Global Unique Identifiers
HTTP	Hypertext Transfer Protocol
IP	Internet Protocol
ISD	Isolation Domain
ISP	Internet Service Provider
LCC	Largest Connected Component
NA	Network Address
NCP	National Chokepoint Potential
NDN	Named Data Network
NDP	Nebula Data Plane
NES-SM	Node-Edge Sampling Method
NSF	National Science Foundation
OSI	Open Systems Interconnection

PIT	Pending Interest Table
PKI	Public Key Infrastructure
RWS	Random-Walk Sampling
R_{AC}	Rewire for Algebraic Connectivity
R_{NP}	Rewire by Network Partition
SCIERA	SCION Education, Research, and Academic
SCION	Scalability, Control and Isolation On next-generation Networks
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TRC	Trust Root Configuration
UI	User Interface
URL	Uniform Resource Locator
XIA	Expressive Internet Architecture
YAML	YAML Ain't Markup Language

List of Figures

List of Tables

Appendix A

Additional Experimental Results

Additional experimental results omitted in Chapter 7 are presented here. Figures A.1 and A.2 show the complete Border Breadth results for each topology across the optimization process.

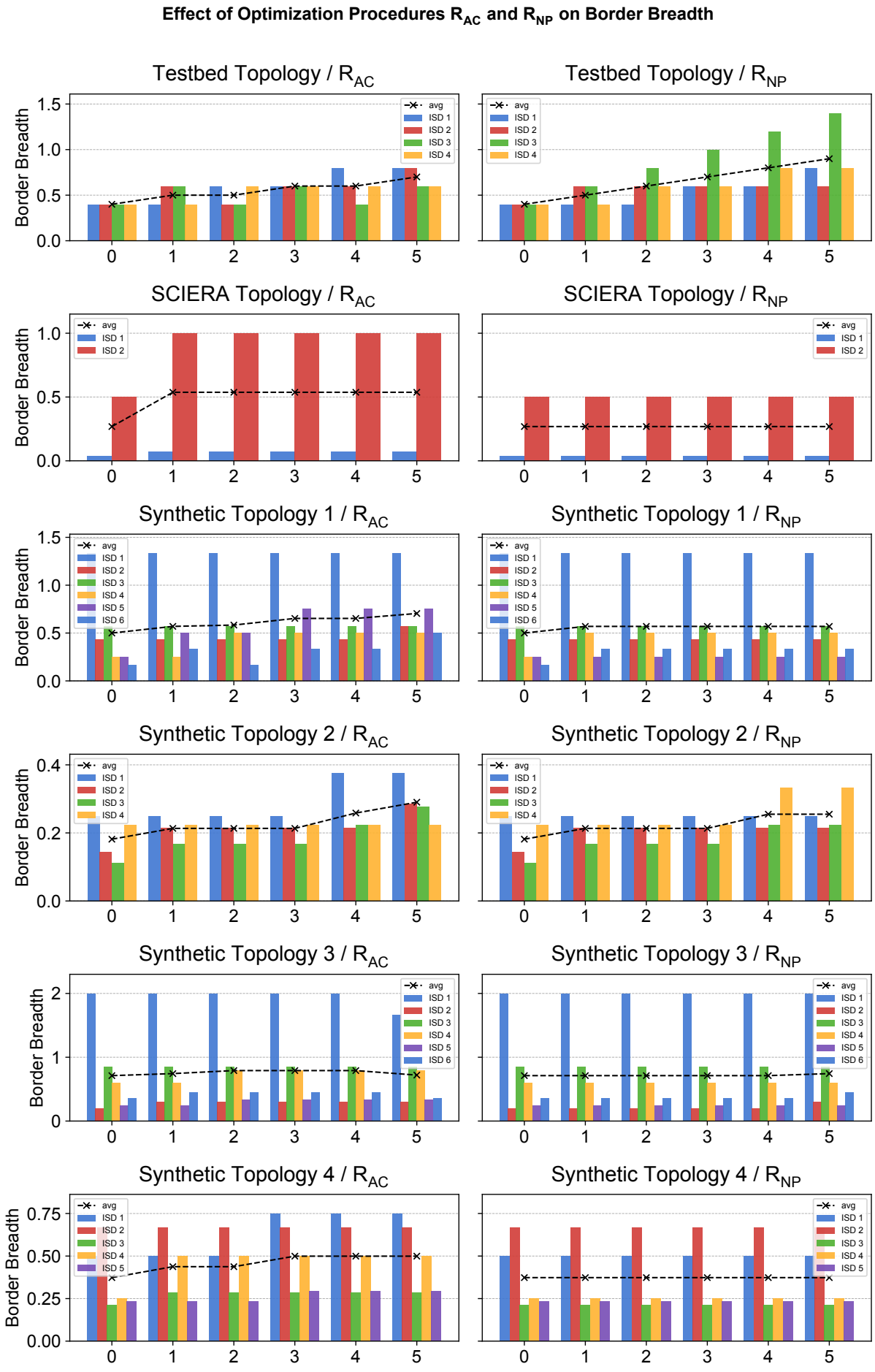

Figure A.1: The Border Breadth results over the course of the optimization processes.

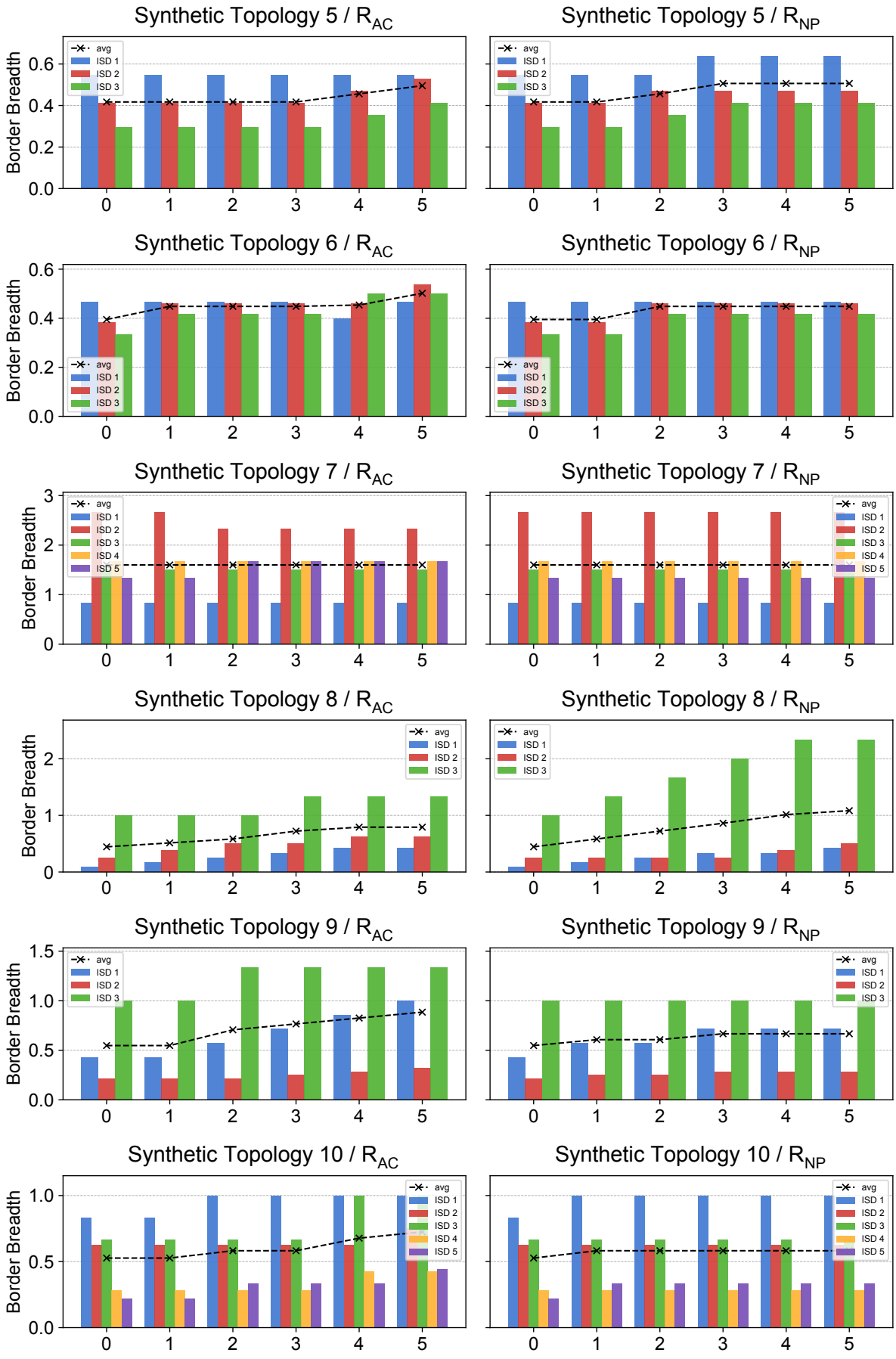

Figure A.2: The Border Breadth results continued over the course of the optimization processes.