WASHINGTON UNIVERSITY IN ST. LOUIS

McKelvey School of Engineering
Department of Computer Science & Engineering

Dissertation Examination Committee:
Roger Chamberlain, Chair
Shantanu Chakrabartty
Ron Cytron
Raj Jain
Ning Zhang

Scaling Quantum Systems: Quantum Networks, Distributed Quantum Computing, and Security

by

Zebo Yang

A dissertation presented to
the McKelvey School of Engineering
of Washington University in
partial fulfillment of the
requirements for the degree
of Doctor of Philosophy

May 2025
St. Louis, Missouri

© 2025, Zebo Yang

Table of Contents

List of Figures vi
List of Tables viii
Acknowledgments ix
Abstract of the Dissertation xi
Chapter 1: Introduction 1
1.2 Quantum Networks 3
1.2.1 Data Qubits and Communication Qubits 3
1.2.2 Quantum Links 5
1.2.3 Quantum Repeaters 8
1.2.4 Quantum Networking Protocols 12
1.3 Distributed Quantum Computing 15
1.3.1 Monolithic and Distributed Execution 15
1.3.2 Circuit Partitioning 18
1.3.3 Qubit Assignment (Qubit Mapping) 19
1.3.4 TeleData and TeleGate 21
1.3.5 Compilation Pipeline for Distributed Quantum Circuits 23
1.4 Quantum Internet Security 25
1.5 Related Work 27
1.5.1 Entanglement Routing 27
1.5.2 Resource Management in DQC 29
1.5.3 Quantum Internet Security 30
1.6 Goals and Contributions 32
1.6.1 Goal 1: Practical and Efficient Entanglement Routing 32
1.6.2 Goal 2: Efficient and Scalable Scheduling for DQC 33
1.6.3 Goal 3: Secure and Resilient Quantum Internet 33
Chapter 2: Asynchronous Quantum Routing 35
2.1 Introduction 35
2.2 Preliminaries 41
2.2.1 Entanglement Swapping 41

2.2.2 Entanglement Purification 42
2.2.3 Global and Local Knowledge of Network Status 43
2.2.4 Finding Disjoint Paths 43
2.2.5 Link Capacity and Desired Fidelity 44
2.3 The Entanglement Routing Problem 45
2.4 Asynchronous Routing Scheme 48
2.4.1 DODAG Protocol 50
2.4.2 Distributed Spanning Tree 55
2.4.3 Time Complexity 59
2.5 Performance Analysis 61
2.5.1 Synchronous Protocol Analysis 62
2.5.2 Asynchronous Protocol Analysis 63
2.5.3 Relative Positions of the Three Nodes in DODAG 70
2.6 Simulation Results 71
2.7 Summary 77
Chapter 3: Multi-tree Quantum Routing 80
3.1 Introduction 80
3.2 Multi-Tree Entanglement Routing 82
3.2.1 Multi-tree Structure 82
3.2.2 Root Nodes Selection 85
3.2.2 Applicability 87
3.3 Simulation Results 88
3.3.1 Simulation Settings 88
3.3.2 Results for Generated Topologies 89
3.3.2 Simulations with Realistic Topologies 91
3.4 Summary 93
Chapter 4: Resource Management in DQC 94
4.1 Introduction 94
4.2 Topologies for DQC 98
4.3 Resource Management Framework for DQC 100
4.4 Optimization and Heuristics 104
4.4.1 Optimization by Mixed Integer Linear Programming 104

4.4.2 Resource Optimization with Non-overlapping Constraint 109
4.4.3 Scheduling by Heuristics 110
4.5 Evaluation 113
4.5.1 Time Complexity 113
4.5.2 Simulation Settings 115
4.5.3 Simulation Results 117
4.6 Summary 120
Chapter 5: Layer-wise Security Framework 121
5.1 Introduction 121
5.2 The Quantum Internet Protocol Stack 125
5.3 Vulnerabilities and Countermeasures 127
5.3.1 Physical Layer 127
5.3.2 Link Layer 135
5.3.3 Network Layer 139
5.3.4 Application Layer 147
5.4 Discussion – Readiness of Quantum Attacks 151
5.5 Summary 154
Chapter 6: Quantum Blockchains 155
6.1 Introduction 155
6.2 Background and Related Work 158
6.2.1 Blockchains and Quantum Blockchains 159
6.2.2 Blockchain-based Identity Applications 161
6.2.3 Related Work 163
6.3 Technical Components 165
6.3.1 Quantum PKI 165
6.3.2 A Chain of Quantum States 171
6.3.3 Primitive Consensus 174
6.4 Quantum Blockchain Identity Framework 175
6.4.1 Roles 175
6.4.2 Quantum Identity Attestations 177
6.4.3 QIAs on Quantum Blockchain 178
6.4.4 Consensus Protocol 180

6.4.5 Attestation and Authentication 182
6.4.6 System Security 185
6.5 Challenges and Limitations 187
6.6 Summary 190
Chapter 7: Conclusion 191
7.1 Summary of Dissertation 191
7.2 Future Directions 193
References 194
Appendix A. Quantum Network Stack 216
Appendix B. DODAG Loop Avoidance 218
Appendix C. GHS Delay Scenarios 220
Appendix D. Mean Path Length Approximation 222
Appendix E. Iterative Approximation 223

List of Figures

Figure 1.1. Generation of Direct-Link Entanglement 6

Figure 1.2. Types of Quantum Repeaters 9

Figure 1.3. Generation of E2E Entanglement in a Quantum Network 13

Figure 1.4. Entanglement Purification 14

Figure 1.5. Two Types of DQC Architectures: (a) Clos and (b) Chiplet 16

Figure 1.6. An Example of a Partitioned Quantum Circuit 18

Figure 1.7. Executing (a) TeleData or (b) TeleGate for a Remote CNOT Gate 21

Figure 1.8. Compilation Pipeline for Distributed Quantum Circuits 24

Figure 2.1. Single-hop Entanglement Generation via Entanglement Swapping 36

Figure 2.2. Topologies in Different Phases and E2E Entanglement Generation 39

Figure 2.3. The Circuit of Entanglement Swapping 42

Figure 2.4. An Example Purification Circuit with Link Capacity Two 42

Figure 2.5. A Link between Alice and Bob with Capacity Two 45

Figure 2.6. Protocol 1. Asynchronous Routing Scheme 49

Figure 2.7. DODAG and its Control Messages 51

Figure 2.8. Protocol 1.1. DODAG Maintenance 54

Figure 2.9. Distributed Spanning Tree by the Modified GHS Algorithm 56

Figure 2.10. Protocol 1.2. Distributed Spanning Tree for Entanglement Routing 59

Figure 2.11. DODAG in a Grid Topology 61

Figure 2.12. The Location Scenarios in a DODAG 65

Figure 2.13. Mean Path Length when Source and Destination must Pass through the Root 68

Figure 2.14. The Triangle Formed by r, s, t 70

Figure 2.15. Simulation Results: Rate vs. Distance 72

Figure 2.16. Simulation Results: Rate vs. p 74
Figure 2.17. Multi-path Simulation Results: Rate vs. p 76
Figure 3.1. A Multi-tree Structure: a DODAG Forest Instance and in a Grid Topology 83
Figure 3.2. Simulation Results for Generated Topologies.....89
Figure 3.3. Simulation Results for Realistic Topologies: SURFnet, ESnet and Internet2 91
Figure 4.1. DCell and BCube Topologies for Quantum Data Centers 99
Figure 4.2. Example of (a) Circuit Partition and (b) Network-aware Quantum Orchestration ...101
Figure 4.3. Algorithm 4.1. Weighted Group Least Resource First (WGLRF) 112
Figure 4.4. Simulation Results of All Methods in DCell and BCube.....117
Figure 4.5. Simulation Results of Fairness and Execution Time.....119
Figure 5.1. Photon Distribution in the Simulation of PNS Attacks 129
Figure 5.2. Phase Randomization in the Trojan-Horse Simulation 132
Figure 5.3. An Infiltration with Entangling Probe 136
Figure 5.4. 100-node Simulations across Various Topologies 141
Figure 6.1. The General Quantum Circuit for a Trapdoor OWQT 168
Figure 6.2. Roles and Interactions on QBIF 179
Figure 6.3. A QBIF Network 180
Figure 6.4. A Consensus Process for Quantum Blockchains.....182
Figure 6.5. An Attestation Process for QBIF.....183
Figure 6.6. Algorithm 6.1. Extracting and Verifying a QIA.....185
Figure 6.7. An Authentication Process for QBIF 185
Figure A.1. 7-layer Stack of Quantum Research 214
Figure B.1. DODAG Greedy Parent Selection 216
Figure C.1. Fragment Combination in the GHS Algorithm.....218

List of Tables

Table 4.1: Symbol Table 103

Table 5.1: Protocol Stack of the Quantum Internet 125

Table 5.2: Readiness of Quantum Attacks 125

Acknowledgments

I would like to express my deepest gratitude to my advisor, Dr. Raj Jain, for his unwavering support, guidance, and invaluable insights throughout my PhD journey. His expertise and encouragement were instrumental in shaping this dissertation.

I am also immensely thankful to the members of my dissertation committee for their constructive feedback and for challenging me to refine my research further. Special thanks to my collaborators and co-authors for their contributions and for making our projects both productive and enjoyable.

My heartfelt appreciation goes to my family and friends for their endless patience, understanding, and encouragement during this demanding journey. Their belief in me has been a constant source of motivation.

Lastly, I gratefully acknowledge the financial support from National Science Foundation (NSF), Cisco Research, Qatar Research Development and Innovation (QRDI) Council, and the Department of Computer Science & Engineering at Washington University in St. Louis, which made this research possible. Thank you all for being an essential part of this journey.

Zebo Yang

Washington University in St. Louis

May 2025

Dedicated to my family.

ABSTRACT OF THE DISSERTATION

Scaling Quantum Systems: Quantum Networks, Distributed Quantum Computing, and Security

by

Zebo Yang

Doctor of Philosophy in Computer Science

Washington University in St. Louis, 2025

Professor Roger Chamberlain, Chair

Recent advancements in quantum computing have opened up new possibilities across various fields, offering significant potential to enhance computation, communication, cryptography, and applications in areas like sensing, medicine, and chemistry. However, the capabilities of individual quantum devices are still limited, and scaling quantum hardware monolithically presents substantial challenges. Interconnecting quantum systems offers a promising alternative by aggregating capabilities across a network, thereby enabling quantum advantages for larger and more practical problems. This interconnection is achieved through quantum networking, which links external quantum systems, and Distributed Quantum Computing (DQC), which connects quantum processors within a system, such as in quantum data centers. As the scale of interconnected systems increases, security concerns associated with their integration also grow.

To advance quantum networking and DQC techniques, numerous models of quantum repeaters and switches have been proposed, each presenting distinct strategies for enabling quantum communication over long distances. Our focus is on repeaters that leverage entanglement generation and swapping, known as “quantum-native” or “first-generation” repeaters. These repeaters face challenges stemming from the probabilistic nature of entanglement generation and

limited coherence times. To address these challenges, this dissertation examines the core principles of quantum networking, DQC, and quantum network security, and presents our contributions to these fields.

In particular, we propose Asynchronous Entanglement Routing (AER) to eliminate the need for synchronized operations required by existing approaches and to preserve unused entanglement links, significantly enhancing the efficiency of End-to-End (E2E) entanglement distribution. AER operates in a distributed manner using only local neighbor information for routing, employing structures such as Destination-Oriented Directed Acyclic Graphs (DODAGs) or distributed spanning trees. By mitigating delays caused by network-wide updates, AER significantly improves practicality for quantum applications. Our results demonstrate that AER achieve a higher upper bound and significantly increase the entanglement rate compared to existing synchronous approaches. The practicality and efficiency gains suggest that AER will have a substantial impact on quantum networks as technology progresses.

We further extend AER to a multi-tree networking scheme to accommodate larger-scale quantum networks. This approach connects multiple DODAGs in a distributed manner, achieving a higher E2E entanglement generation rate in a practical and asynchronous fashion, while still relying solely on the local knowledge of each node's entanglement links. Testing on both generated and realistic topologies indicates that the multi-tree approach outperforms single-tree and synchronous routing schemes in terms of entanglement rate and efficiency. This work highlights the effectiveness of asynchronous routing strategies for diverse quantum network topologies and presents a superior routing solution.

In addition to interconnecting external quantum systems, the need for linking Quantum Processing Units (QPUs) within a single system has also grown, driving the expansion of DQC architectures. In a quantum data center with multiple QPUs distributed across different racks, executing remote gates requires entanglement between distinct QPUs. However, generating this entanglement often leads to congestion and resource contention. To address this, we propose a resource management framework that maximizes fidelity-guaranteed throughput while maintaining dependencies. We formulate the problem as a Mixed-Integer Linear Programming (MILP) model for benchmarking and introduce efficient heuristic scheduling algorithms. Simulations show that these heuristics closely approximate the solver's results, demonstrating their practicality for managing remote gates and improving overall performance.

Despite the enhanced security that quantum networks offer over traditional systems, the quantum internet still faces unique security challenges crucial for ensuring its Confidentiality, Integrity, and Availability (CIA). We examine these challenges by analyzing vulnerabilities and potential mitigation strategies across different layers of the quantum internet, including the physical, link, network, and application layers. We assess the severity of potential attacks and evaluate the effectiveness of corresponding countermeasures, integrating both classical and quantum approaches. The findings highlight the dynamic nature of security risks and emphasize the need for adaptive security measures. It underscores the need of continuous exploration into the security aspects of the quantum internet to enhance its robustness and facilitate its widespread adoption.

We further explore quantum-era cryptographic applications through a theoretical analysis of quantum blockchains for decentralized identity authentication. It includes a proposed conceptual framework, a review of supporting technical evidence, and an evaluation of its core components, feasibility, effectiveness, and limitations.

Chapter 1: Introduction

Recent advancements in quantum computing have opened up new possibilities across various fields, offering significant potential to enhance computation, communication, cryptography, and applications in areas such as sensing, medicine, and chemistry [1], [2], [3], [4], [5], [6]. These developments have been propelled by the unique properties of quantum mechanics, including superposition and entanglement [7], [8], [9], which enable capabilities far beyond those of classical systems. However, the capabilities of individual quantum devices remain limited due to factors such as coherence times, error rates, and the complexity of scaling quantum hardware monolithically [10], [11], [12]. Overcoming these challenges has become a primary focus of current quantum research [13], [14], [15], [16].

Interconnecting quantum systems presents a promising alternative to monolithic scaling by aggregating capabilities across a network. This approach leverages quantum networking [17], [18], which enables communication between separate quantum systems, and Distributed Quantum Computing (DQC) [19], [20], which integrates multiple quantum processors within a unified framework, such as quantum data centers. By distributing computational tasks and facilitating resource sharing, these interconnected systems aim to harness the full potential of quantum advantages for solving larger and more practical problems that are currently infeasible for individual quantum devices.

As the scale of interconnected quantum systems continues to expand, the complexity of managing resources, maintaining coherence, and optimizing communication increases significantly. In this context, quantum repeaters and switches play a crucial role in extending the range of quantum communication by enabling entanglement generation and swapping over long distances [21], [22],

[23]. However, these mechanisms introduce additional challenges, including probabilistic entanglement generation, resource contention, and the need for efficient routing protocols that can adapt to the dynamic nature of quantum networks [24], [25].

Moreover, the integration of multiple quantum systems raises substantial security concerns. Ensuring the Confidentiality, Integrity, and Availability (CIA) of information in quantum networks requires addressing vulnerabilities across various layers, from the physical to the application level [26], [27]. The emergence of the quantum internet, which envisions a global network of quantum systems, has intensified the demand for advanced security protocols capable of mitigating potential threats, such as eavesdropping and entanglement manipulation [28], [29]. As a result, designing robust security frameworks for quantum networks has become a vital area of research.

This dissertation aims to address these challenges by proposing new techniques for routing, resource management, and security in quantum networks and DQC architectures. It explores both the theoretical foundations and practical implementations of these techniques, providing a novel set of techniques and a comprehensive analysis of their effectiveness in enhancing the scalability, efficiency, and security of interconnected quantum systems. In doing so, this work seeks to contribute to the advancement of scalable and secure quantum networks that can support a broad range of applications, paving the way for the future quantum internet.

The remainder of this chapter presents the fundamentals of quantum networks, DQC, and quantum network security, followed by an overview of related work, the goals and the contributions of this dissertation. These contributions include the development of an asynchronous quantum routing scheme, resource management in DQC, a layer-wise security framework for the quantum internet, and an exploration of quantum applications such as quantum blockchains.

1.2 Quantum Networks

Quantum networks represent a transformative step beyond classical networking by leveraging the principles of quantum mechanics to enable secure communication, secret key distribution, and novel applications that classical networks cannot support [30]. The architecture of quantum networks consists of several key components that work together to establish, manage, and utilize entanglement between distant nodes efficiently [16], [31]. This section introduces the core elements, including communication qubits, quantum links, quantum repeaters, and quantum networking protocols, as well as additional components such as quantum memories and quantum switches that play vital roles in the network's performance and scalability.

1.2.1 Data Qubits and Communication Qubits

There exist various types of qubits, each with distinct advantages and challenges based on their intended applications in quantum networks. The selection of qubit type plays a crucial role in determining the performance, scalability, and practicality of both quantum communication and computation. Some qubits are optimized for processing and storing information within quantum processors, known as *data qubits*, while others are designed for transmitting information across distances, referred to as *communication qubits* [14], [32]. This subsection presents the different types of qubits, including superconducting, photonic, trapped-ion, and topological qubits, and discusses their typical roles as either data qubits or communication qubits.

Superconducting qubits, which are built on Josephson junctions [33], represent one of the most advanced and widely utilized types of qubits in quantum computing, as demonstrated by the quantum processors developed by IBM and Google [34], [35], [36]. They offer fast gate operations and are compatible with existing semiconductor fabrication techniques, making them a leading

candidate for building quantum processors. However, their coherence times are relatively short, which limits their suitability for long-distance communication. Superconducting qubits are primarily used as data qubits within quantum processors to perform computations.

Photonic qubits are the primary choice for communication qubits due to their ability to travel long distances with minimal decoherence through optical fibers [37], as demonstrated by companies such as Xanadu and PsiQuantum [38], [39]. Typically encoded using polarization, time-bin, or path-based methods, photonic qubits facilitate the transmission of quantum information between remote nodes. Their compatibility with existing optical infrastructure makes them indispensable for scalable quantum networks.

Trapped-ion qubits are based on individual ions confined and manipulated using electromagnetic fields [40], as exemplified by IonQ's quantum processors [41]. They are known for their long coherence times and high-fidelity gate operations, making them highly reliable for quantum computations. Although gate speeds are slower compared to superconducting qubits, trapped-ion qubits offer exceptional stability, which is advantageous for both data and communication purposes. Their compatibility with photonic interconnects also makes them a promising candidate for hybrid quantum networks.

Topological qubits, based on anyons and topological phases of matter [42], are still largely theoretical but hold great promise due to their intrinsic resistance to local noise and decoherence [43], [44]. Microsoft has been actively working on developing topological qubits and recently announced the creation of the first quantum chip powered by a topological core architecture [45]. By encoding information in the global properties of a system, topological qubits offer a natural form of error protection, which could significantly reduce the overhead required for quantum error

correction. This resilience makes them a potential candidate for both data and communication qubits in large-scale quantum networks.

1.2.2 Quantum Links

Quantum links serve as the foundational communication channels in quantum networks, enabling the transfer of quantum states between spatially separated nodes, either for direct qubit transmission or for the establishment of entangled pairs across distant locations. Unlike classical links that carry bits through electrical or optical signals, quantum links use quantum states, which are highly sensitive to environmental disturbances and cannot be copied due to the no-cloning theorem [46]. Therefore, establishing reliable and high-fidelity quantum links poses significant technical challenges, particularly over long distances where decoherence and photon loss become pronounced.

Quantum links can be broadly categorized into two types: direct transmission links and links dedicated to entanglement generation. Direct transmission links involve the physical transmission of qubits, often in the form of photons, through optical fibers or free-space channels [47], [48]. These are commonly used in protocols like BB84 [49] for Quantum Key Distribution (QKD), where quantum states are transmitted directly from sender to receiver. However, they are inherently limited by losses in the transmission medium, especially over fiber distances exceeding a few hundred kilometers [50], [51].

Links dedicated to entanglement generation, in contrast, enable the establishment of entangled qubit pairs between two remote parties. This process, referred to as *direct-link entanglement* generation, is a core capability of quantum networks. Several methods exist for generating entangled pairs between nodes. A common approach involves creating entangled photon pairs at

a central source and distributing one photon to each party (e.g., Alice and Bob). Alternatively, entanglement can be generated at one node (e.g., Alice), with one qubit converted into a photon and transmitted to the other node (e.g., Bob) [52], as shown in Figure 1.1.

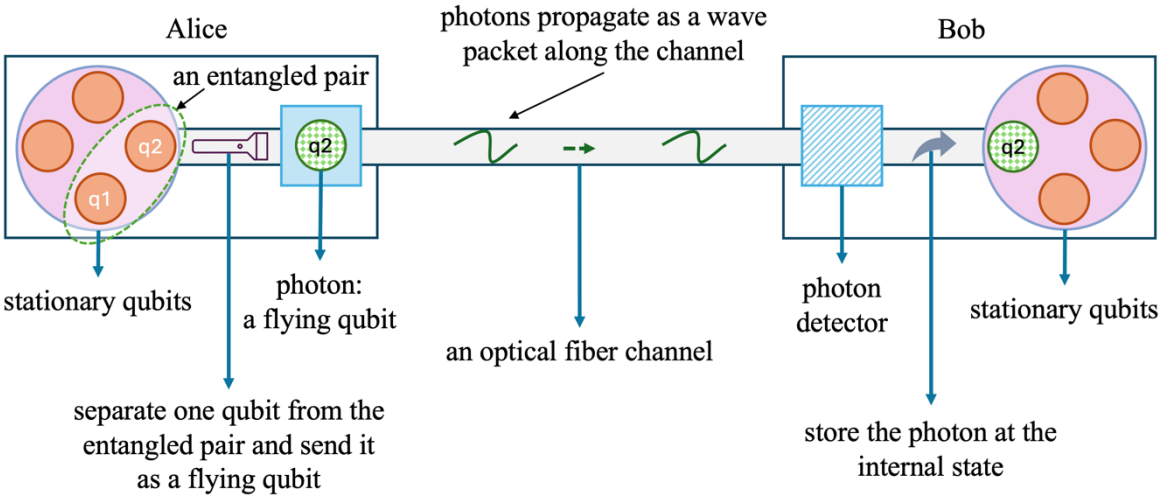


Figure 1.1. *Generation of Direct-Link Entanglement.*

In Figure 1.1, Alice wants to create a direct-link entanglement with Bob. Alice then generates an entangled pair (e.g., q1 and q2) using her *stationary qubits* stored in quantum memory, which, in this context, function as data qubits, since entanglement generation typically requires local qubit interaction [53]. Stationary qubits refer to localized qubits, usually situated at a network node (e.g., within Alice's quantum processor or memory). Depending on the system's needs, these qubits can serve various roles: as data qubits (storing algorithmic or message-related information), memory qubits (temporarily holding quantum states), or ancilla qubits (supporting gate operations or error correction) [54].

Let us assume the stationary qubits in Figure 1.1 are implemented using trapped ions. These can be efficiently converted into photonic qubits, a well-established technique, especially in the context of quantum communication and DQC [55], [56]. To establish a direct-link entanglement,

Alice converts one qubit from an entangled pair (e.g., q2) into a photonic qubit. This flying qubit, typically regarded as a communication qubit in quantum networks, can then be transmitted through an optical fiber channel to Bob, who is located at the other end. Bob, equipped with a photon detector, receives q2 and stores it in his quantum memory. As a result, a direct-link entanglement between Alice and Bob is established. This shared entanglement can then be used for various communication tasks such as QKD, quantum teleportation, or entanglement swapping to further distribute entanglement to nodes that do not share a direct connection. Such setups have been experimentally demonstrated and repeated across multiple platforms worldwide [52], [57], [58].

This process of generating direct-link entanglement is inherently probabilistic and highly sensitive to loss and timing mismatches. To complement this, heralding signals (classical messages that indicate successful entanglement events) are commonly employed to confirm the establishment of a quantum link. Once a link is successfully heralded, the entangled states can be temporarily stored in quantum memories at the endpoints, awaiting further operations such as entanglement swapping or quantum teleportation. Consequently, any functional quantum link inherently relies on an accompanying classical communication channel. That is, the quantum networks being developed today are, in essence, hybrid quantum-classical networks. This design philosophy reflects the broader vision of quantum computing: not to replace the existing internet infrastructure, but to augment it by enabling capabilities such as quantum-secured communication and quantum computational supremacy [59], [60].

In addition to optical fiber, free-space quantum links have also attracted significant attention, particularly in the context of satellite-based quantum communications [61], [62], [63], which has enabled entanglement distribution over distances spanning thousands of kilometers [64], [65]. These free-space links complement terrestrial fiber networks by circumventing ground-level and

atmospheric obstructions, thereby facilitating global-scale quantum connectivity. Ultimately, whether implemented via fiber, free-space channels, or satellite relays, quantum links form the foundational infrastructure of quantum networks. Their performance, characterized by metrics such as entanglement fidelity, link success probability, coherence time, and channel loss, plays a critical role in determining the overall efficiency, scalability, and robustness of quantum communication systems.

In this dissertation, unless otherwise specified, "quantum links" refers to links used for the generation of entangled pairs between remote nodes, thereby enabling entanglement distribution.

1.2.3 Quantum Repeaters

Quantum repeaters address the challenges of photon loss and decoherence over long-distance quantum links by segmenting the communication path into shorter sections [18], [21]. Following the two types of quantum links, there are two corresponding types of quantum repeaters: *one-way repeaters* to forward qubits and *two-way repeaters* to generate remote entanglement (also known as "quantum-native" or "first-generation" repeaters). In one-way (or qubit-forwarding) repeaters, quantum information is transmitted from Alice to Bob through the repeater in a single direction, using quantum error correction to preserve the qubit without requiring classical feedback [22], [66], [67], [68], [69]. In two-way (or entanglement-swapping) repeaters, Alice and Bob each establish entanglement with the repeater, which then performs entanglement swapping, e.g., Bell State Measurement (BSM), to create End-to-End (E2E) entanglement, requiring classical communication in both directions [21], [23], [70], [71], [72]. Additionally, there are ongoing efforts focused on QKD trusted relays, which are sometimes also referred to as quantum repeaters [50], [73], [74], [75]. In QKD trusted relays, separate quantum key distribution sessions are

conducted between Alice and the repeater and between the repeater and Bob, with the repeater combining and relaying the key under the assumption that it is fully trusted. The different types of repeaters are summarized in Figure 1.2.

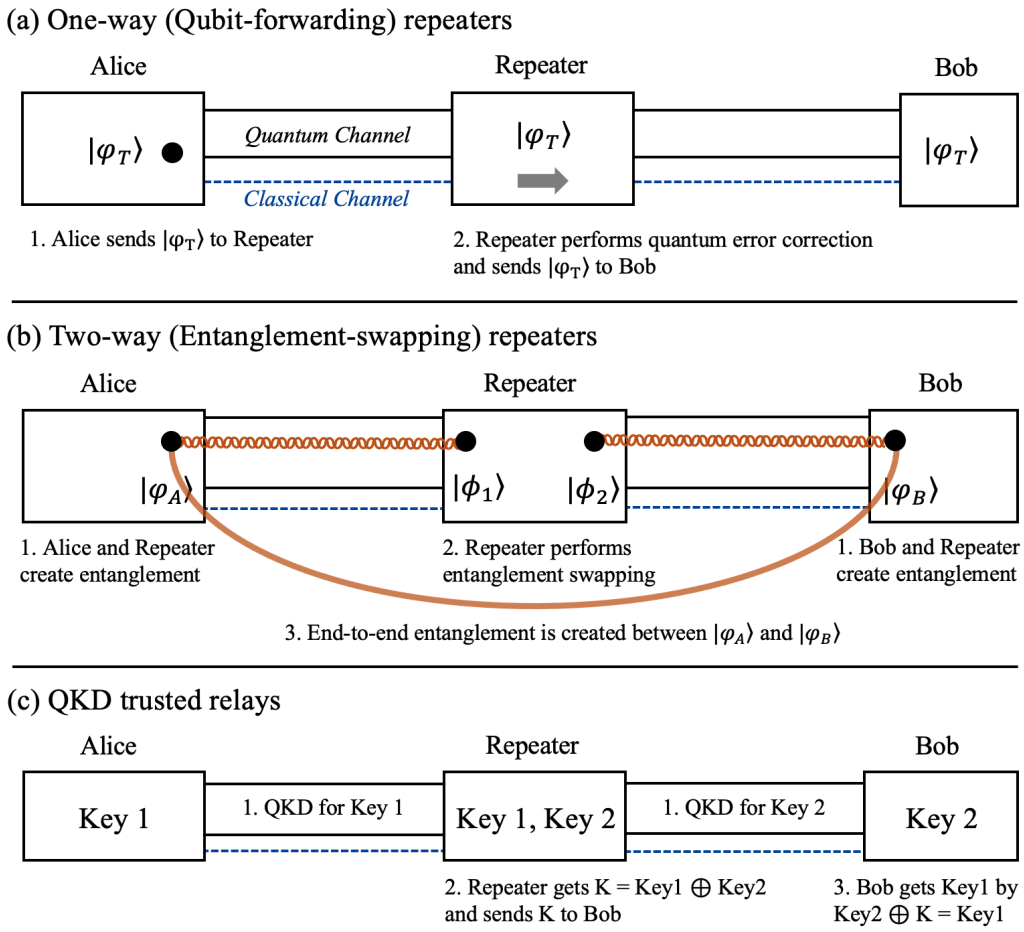


Figure 1.2. *Types of Quantum Repeaters.*

One-way quantum repeaters, based on direct transmission links, aim to minimize reliance on back-and-forth classical communication, thereby reducing latency, though at the cost of more demanding technological requirements. Quantum information is pre-encoded using quantum error correction techniques, such as surface codes or concatenated codes [76], [77], into logical qubits that are resilient to noise and loss. These encoded states are transmitted unidirectionally from the source to the destination through a series of repeater nodes. Each repeater operates using fixed,

predefined procedures without waiting for classical confirmation, simply forwarding the quantum state along the chain. At the destination, or at designated points along the path, errors are identified and corrected through syndrome measurements, all without interrupting the forward flow of communication.

One-way repeaters have no round-trip delays caused by classical communication, enabling faster transmission and making them well-suited for real-time applications such as the quantum internet and large-scale DQC. They also offer greater scalability for global quantum networks. However, this approach demands high-fidelity quantum error correction and long-lived quantum memories, requiring each repeater node to support reliable error correction operations. As a result, the hardware infrastructure is more complex compared to that of two-way repeater architectures.

A two-way quantum repeater, based on links dedicated to entanglement generation, is termed "two-way" because it relies on classical communication in both directions (between the sender and receiver) throughout the processes of entanglement generation and entanglement swapping. When neighboring nodes (e.g., Node A and Node B) attempt to establish entanglement, they must exchange classical messages to verify success. Similarly, during entanglement swapping to extend entanglement over longer distances, classical communication is required to coordinate and confirm measurement outcomes. Since quantum operations, such as measurements, are inherently probabilistic, classical feedback is essential for validating results and applying necessary conditional operations. This bidirectional communication ensures the successful generation of E2E entanglement across the network. Two-way repeaters create local entanglement segments and use entanglement swapping, supported by quantum memories that temporarily store quantum states, to extend this entanglement. Their efficiency is critical for enabling reliable long-distance quantum communication.

Two-way quantum repeaters are more aligned with current experimental capabilities than the more technologically demanding one-way, error-corrected designs. They have already been demonstrated in laboratory settings using platforms such as atomic ensembles, nitrogen-vacancy (NV) centers, and trapped ions [78], [79], [80]. Unlike one-way repeaters, two-way architectures do not require full quantum error correction, making them more tolerant of lower-fidelity qubits. Their implementation is relatively straightforward, often relying on quantum memories and BSM [21]. Additionally, two-way protocols can incorporate entanglement purification to improve the fidelity of shared entanglement by discarding lower-quality pairs, an important feature for high-fidelity applications like quantum teleportation and QKD. In general, entanglement purification is less resource-intensive than full quantum error correction [81].

However, a key limitation of two-way repeaters is their reliance on round-trip classical communication after each entanglement attempt. This introduces latency, which becomes a major bottleneck over long distances. As entanglement generation is inherently probabilistic, multiple attempts may be required before success, and each attempt must wait for classical confirmation, further reducing throughput. These delays, combined with memory decoherence, make two-way repeaters less scalable for global networks unless mitigated by advanced engineering solutions such as high-coherence memories or parallelized entanglement generation. Despite being simpler than one-way designs, two-way repeaters still demand quantum memories with sufficient coherence times to preserve quantum states during these communication delays.

In this dissertation, unless stated otherwise, the term “quantum repeaters” refers to two-way quantum repeaters that enable entanglement distribution.

1.2.4 Quantum Networking Protocols

Current classical networking protocols can, in principle, be directly applied to one-way repeater schemes; however, their practicality is significantly limited by the substantial qubit overhead required for quantum error correction. In this dissertation, we focus on quantum networking protocols specifically designed for networks built on two-way repeater architectures. Similar to how early wireless sensor networks shifted from general-purpose Internet protocols to specialized, resource-efficient solutions [82], [83], [84], quantum networks are now undergoing a comparable transformation, toward more customized, hardware-conscious protocol designs.

Unlike classical networks, where data can be freely copied and retransmitted, quantum information is subject to the no-cloning theorem, which prohibits duplicating unknown quantum states [46]. This fundamental principle introduces unique constraints on quantum data transmission and necessitates the development of entirely new networking paradigms. Additionally, photon loss and decoherence significantly reduce both the reliability and transmission range of quantum information. As quantum networks expand beyond simple linear chains of repeaters, it becomes essential to determine optimal routing paths (that is, practical paths that maximize entanglement throughput) between any two nodes. Quantum networking protocols are therefore designed to address these challenges by orchestrating entanglement generation, distribution, and coordination across multiple nodes with the help of quantum repeaters.

Two-way quantum repeaters enable the extension of entanglement over long distances by segmenting the communication path into shorter links and using entanglement swapping to “stitch together” an E2E entangled connection. Quantum networking protocols are responsible for identifying optimal paths through the network, allowing intermediate repeaters to perform

entanglement swapping and establish entanglement between the end nodes. These protocols are organized into layered sets of rules and operations that govern the interaction between quantum and classical components of the network.

In the setting of two-way repeaters, these protocols manage several interdependent tasks. First, they enable the generation of entanglement between neighboring nodes through quantum channels, such as optical fibers. Since entanglement generation is probabilistic, classical communication is used to exchange measurement outcomes and control messages to confirm successful generation and synchronize subsequent actions. Once entanglement is established across adjacent links, the protocol initiates entanglement swapping operations at the repeater nodes, as shown in Figure 1.3(a). Throughout these processes, quantum memory plays a critical role by storing qubits while classical messages are exchanged, and other operations are completed [85], [86]. To ensure reliability and scalability, the protocol must also manage errors and optimize the use of limited resources, such as available qubits, memory lifetimes, and channel capacities.

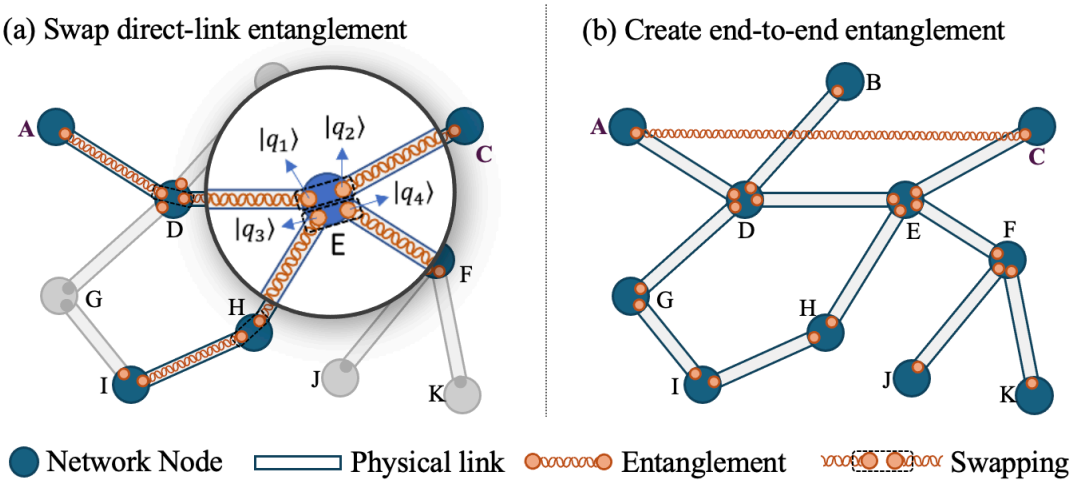


Figure 1.3. *Generation of E2E Entanglement in a Quantum Network.*

Together, these coordinated operations enable quantum networking protocols to fulfill their primary function: the reliable establishment of E2E entanglement between distant nodes over a distributed network of quantum repeaters, as shown in Figure 1.3(b).

Quantum networking protocols built on two-way repeaters represent a foundational step toward building functional quantum networks. While not yet scalable to global distances without further innovation, they provide a feasible and experimentally grounded framework for quantum networks today. By coordinating Local Operations and Classical Communications (LOCC) [87], these protocols enable the delivery of E2E entanglement. The goal is to identify paths with the highest E2E entanglement generation rate. As quantum hardware improves, these protocols will continue to evolve to support more efficient, low-latency, and large-scale quantum communication.

In cases where high-fidelity entanglement is required, the protocol may incorporate entanglement purification, which improves quality by consuming multiple lower-fidelity pairs [88], [89]. As shown in Figure 1.4, two rounds of purification are applied to three raw entangled pairs to produce a single entangled pair with higher fidelity.

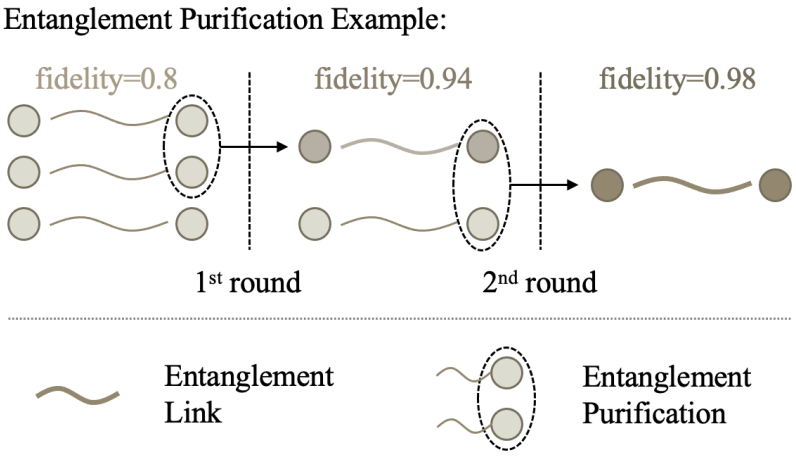


Figure 1.4. *Entanglement Purification.*

1.3 Distributed Quantum Computing

As quantum processors scale in qubit count and complexity, the limitations of monolithic architectures become increasingly evident. Factors such as limited qubit connectivity, restricted chip size, and coherence time constraints make it difficult to build large, universal quantum processors on a single physical device [90], [91]. DQC offers a promising approach to overcome these limitations by interconnecting multiple quantum processors, often referred to as Quantum Processing Units (QPUs), into a networked system capable of collectively executing quantum circuits. DQC enables resource sharing, modular scalability, and potentially higher fault tolerance through spatial separation of quantum logic [92]. This section explores the fundamental concepts and architectural considerations of DQC, including execution models, remote operations, circuit decomposition, and qubit assignment strategies.

1.3.1 Monolithic and Distributed Execution

In monolithic quantum systems, all qubits are integrated within a single processor, allowing direct application of quantum gates between physically connected qubits. However, as the number of qubits increases, several challenges emerge that hinder scalability and performance. One of the primary limitations arises from the physical layout of the system. For example, superconducting processors often arrange qubits in planar architectures, such as square or heavy-hexagonal lattices, where each qubit is typically connected to a limited number of neighbors [93]. IBM's Falcon processors, for instance, employ a heavy-hex connectivity graph optimized for their cross-resonance two-qubit gates [94]. While this design facilitates certain operations, it restricts the ability to implement algorithms requiring higher connectivity, necessitating additional operations like SWAP gates to move qubits into interacting positions [95]. These extra operations increase circuit depth and the potential for errors.

Moreover, as qubit density rises, unintended interactions between qubits, known as crosstalk, become more prevalent [96], [97], [98]. Crosstalk can lead to errors during quantum operations, as qubits inadvertently influence each other's states. Managing crosstalk requires careful calibration and isolation strategies, which become increasingly complex as the system scales [99]. Additionally, scaling up monolithic QPUs demands more intricate control wiring and electronics to manage a larger array of qubits. Integrating extensive control circuits and wiring within a single chip presents significant challenges, including increased spatial constraints and potential signal interference. These factors complicate the design and fabrication processes, limiting the scalability of monolithic architectures [100].

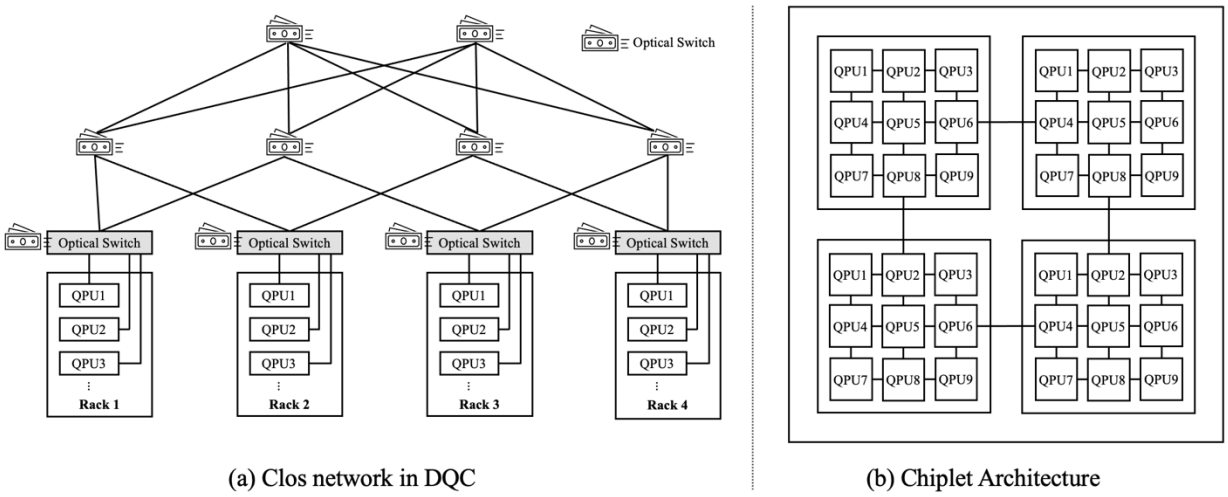


Figure 1.5. *Two Types of DQC Architectures: (a) Clos and (b) Chiplet.*

Due to these challenges, alternative architectures, such as modular or DQC systems, are being explored to facilitate scalability and enhance performance. These approaches aim to interconnect multiple QPUs, each with manageable qubit counts and control requirements, to collectively perform complex computations. In DQC, a quantum circuit is divided among multiple physically separated QPUs, interconnected through quantum communication channels. This architecture enables the execution of complex quantum algorithms by leveraging the combined computational

power of multiple QPUs. Figure 1.5 illustrates examples of DQC architectures, including a Clos network in which QPUs are interconnected via direct links and optical switches [101], and a chiplet-based architecture where QPUs are directly linked in a grid-like topology [102].

To support distributed execution, the first step involves decomposing complex quantum algorithms through a process known as circuit partitioning. This entails breaking down a global quantum circuit into smaller subcircuits, with each either assigned to a specific QPU or executed collaboratively across multiple QPUs through remote gates. The primary objective is to minimize inter-QPU communication while maintaining efficient circuit execution, considering gate locality, qubit interaction patterns, and hardware constraints [103]. Once the circuit is partitioned, operations involving qubits located on different QPUs require entanglement-assisted protocols, such as quantum teleportation. Further details on such protocols are introduced in Subsection 1.3.4.

Despite its promise, distributed execution presents several challenges that must be addressed to achieve scalable and efficient DQC. One major concern is communication latency, as the probabilistic nature of remote entanglement generation often leads to delays, typically longer than local gate operations. Another critical issue is the fidelity of remote operations; maintaining high-quality entanglement between physically separated QPUs is difficult due to factors such as decoherence, noise, and imperfections in entanglement distribution. Furthermore, resource coordination becomes increasingly demanding as the network grows. Effectively managing entangled pairs and quantum memory across multiple QPUs is essential to avoid performance bottlenecks and requires the use of advanced strategies like adaptive scheduling and buffering.

1.3.2 Circuit Partitioning

In DQC, a complex quantum circuit must be partitioned into multiple sub-circuits to facilitate distributed execution. A perfectly distributable quantum circuit is one that can be divided into independent sub-circuits that either do not interact or interact only minimally with one another. In such cases, each sub-circuit can be assigned to a QPU, allowing each QPU to contribute independently to the overall computation without incurring communication overhead. However, this is rarely the case.

In a typical quantum circuit, such as the Quantum Fourier Transform (QFT), each qubit tends to interact strongly with all other qubits [104]. Distributing such a circuit requires the use of a quantum compiler to determine the optimal partitioning, aiming to minimize the number of gates applied to qubits stored on different devices [105], [106]. In other words, circuit partitioning aims to divide the global complex circuit in a way that minimizes inter-QPU communications while balancing the computational load across nodes. This is analogous to graph partitioning in classical distributed systems [107], where minimizing the cut size (i.e., the number of remote gates) is crucial for efficiency.

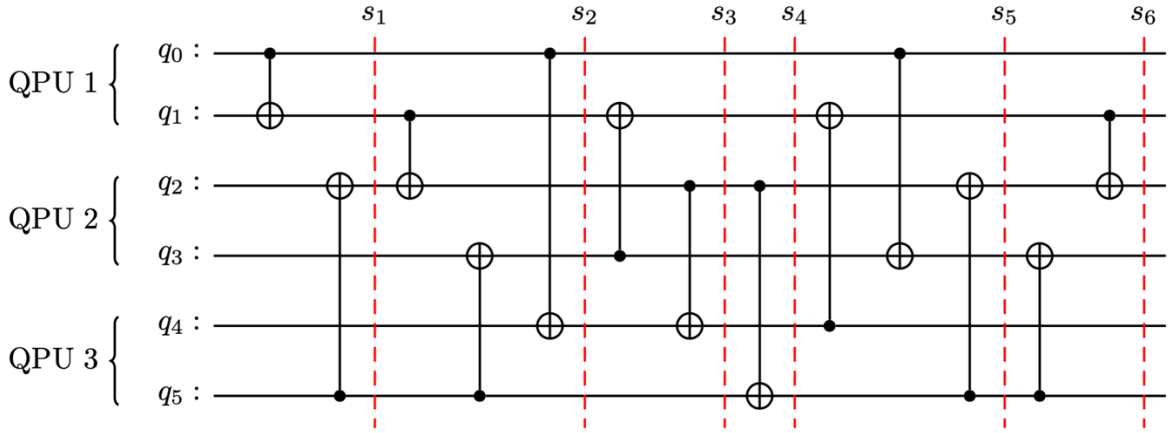


Figure 1.6. *An Example of a Partitioned Quantum Circuit.*

In this subsection, we focus on partitioning a quantum circuit such that the execution order of gates within a sub-circuit is flexible, while the execution order of the sub-circuits themselves must be preserved. As shown in Figure 1.6, an example quantum circuit is divided into six sub-circuits (i.e., s_1 to s_6). Each sub-circuit contains multiple quantum gates, including remote operations (e.g., CNOT gates) between different QPUs. The CNOT gates within a sub-circuit can be executed in any order because no single data qubit is shared by more than one gate within the sub-circuit. Each remote gate requires a certain number of E2E entangled pairs, with higher quantities preferred to meet fidelity requirements. This setup enables the management of batches of E2E entangled pair requests, either from groups of remote gates within a single sub-circuit or spanning multiple sequential sub-circuits with dependency relationships. The goal is to maximize the number of satisfied requests within a given time interval.

Effective partitioning strategies must consider gate locality, qubit dependencies, and physical limitations of each QPU. Techniques may include greedy heuristics, graph-based algorithms, or optimization-based formulations, e.g., Mixed Integer Linear Programming (MILP). Additionally, partitioning must be aware of dynamic conditions, such as the availability of communication links or constraints imposed by quantum memory coherence times.

1.3.3 Qubit Assignment (Qubit Mapping)

Once the circuit is partitioned, the next critical step is qubit assignment, which involves mapping each logical qubit in the circuit to a physical qubit on a designated QPU [14]. This mapping must carefully consider various hardware-level constraints, including the physical connectivity between qubits, the fidelity of gate operations for individual qubits, and crosstalk that may occur when operations on one qubit unintentionally affect others. Additionally, in a distributed setting, the

inter-node communication cost must be factored in, as remote gates require entanglement generation and classical coordination across QPUs.

An inefficient qubit assignment can significantly degrade system performance. For instance, if frequently interacting qubits are placed far apart, either on different QPUs or on non-adjacent physical qubits within the same QPU, additional operations are needed to bring them into proximity. In monolithic systems, this is typically done using SWAP gates, which exchange the quantum states of two qubits [95]. While functional, SWAP gates introduce additional gate depth, increase the number of operations, and thereby amplify decoherence and error accumulation. In distributed systems, a poor mapping may result in excessive teleportation, which not only increases latency but also consumes more entangled pairs and classical communication bandwidth.

Efficient qubit assignment seeks to minimize these overheads by strategically placing logical qubits with frequent interactions close to one another, either physically in hardware or logically through entangled links. Some approaches use heuristic or optimization-based methods (e.g., graph partitioning) to balance load, minimize remote gate operations, and reduce SWAP requirements, thereby improving the overall fidelity and execution time of the distributed quantum circuit [108], [109]. Nonetheless, static assignment may not always be sufficient. In DQC, dynamic factors such as qubit decoherence, gate fidelity drift, or even temporary link failures between QPUs can degrade the efficiency of a precomputed qubit layout. To address this, some systems implement dynamic qubit remapping, which allows qubit assignments to be updated at runtime in response to system state changes [110], [111]. For example, if a QPU experiences increased noise or memory degradation, certain qubits can be remapped to alternate locations with better performance characteristics.

1.3.4 TeleData and TeleGate

As outlined in Subsection 1.2.1, qubits within a QPU are typically categorized as either communication qubits or data qubits in the context of DQC and, more broadly, quantum networks. This distinction is fundamental for enabling the generation and distribution of E2E entangled states, which are critical for executing remote quantum gates across distributed systems. E2E entanglement facilitates remote gate operations using approaches such as TeleData and TeleGate [112], which allow for either the transfer of quantum states between QPUs prior to gate execution or the direct application of gates across spatially separated QPUs.

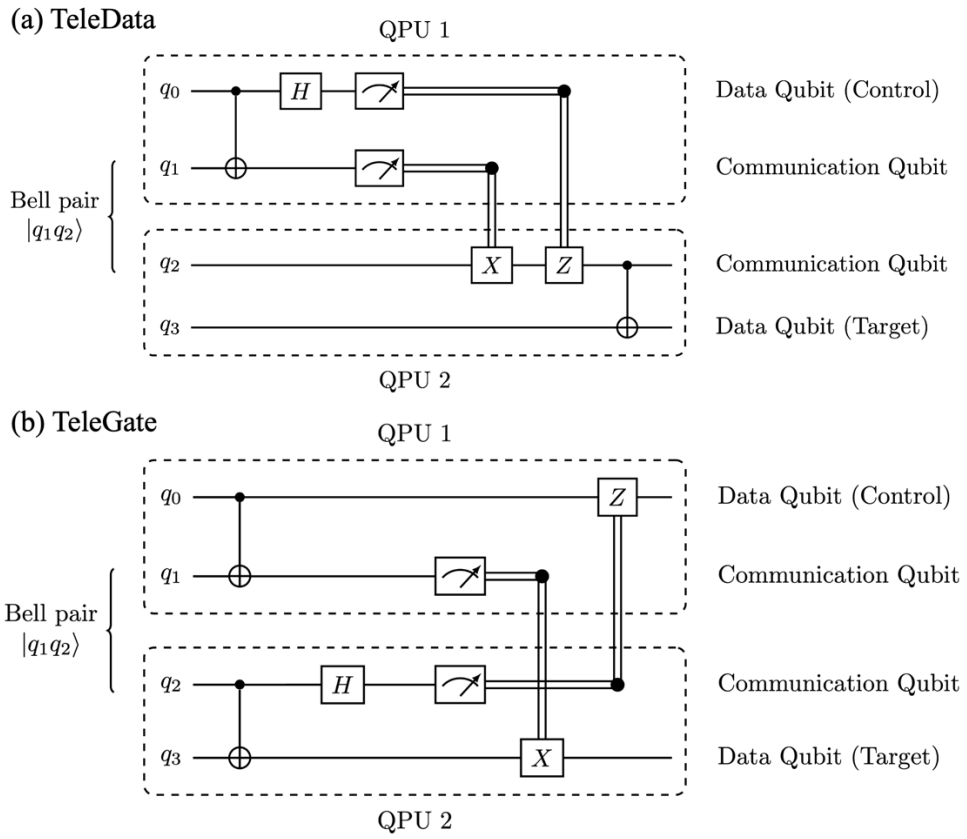


Figure 1.7. *Executing (a) TeleData or (b) TeleGate for a Remote CNOT Gate between QPUs 1 and 2.*

With TeleData, quantum information stored in a data qubit at one QPU, such as $|\varphi\rangle$ stored in q_0 at QPU 1 in Figure 1.7(a), is teleported to a communication qubit at another QPU, such as q_2 at

QPU 2. This teleportation utilizes the E2E entanglement between QPUs, specifically the Bell pair $|q_1 q_2\rangle$ in Figure 1.7(a). Once $|\varphi\rangle$ is teleported to q_2, any operations previously requiring interaction between q_0 in QPU 1 and q_2 in QPU 2 can be performed locally on QPU 2. In the Figure 1.7(a) example, a remote CNOT gate is executed with the data qubit q_0 in QPU 1 as the control qubit and the data qubit q_3 in QPU 2 as the target qubit.

In contrast, TeleGate enables direct gate operations between data qubits in different QPUs without teleportation, provided E2E entanglement exists between the QPUs. Let us consider the same example of implementing a remote CNOT gate using TeleData, where q_0 in QPU 1 serves as the control qubit and q_3 in QPU 2 acts as the target qubit. As shown in Figure 1.7(b), this remote gate can be executed by performing two local CNOT gates: one between the data and communication qubit in each QPU, followed by a conditional operation on the data qubit based on the measurement outcome of the remote communication qubit.

Each method has its own advantages and limitations. TeleData is beneficial when a quantum state must be fully transferred to another QPU for processing, enabling seamless local execution of subsequent operations. However, it incurs extra latency and overhead due to the teleportation process. TeleGate, on the other hand, avoids state relocation by facilitating direct remote operations. However, implementing TeleGate can be more challenging as it demands precise synchronization and high-fidelity entanglement for accurate gate execution. The decision between TeleData and TeleGate depends on the specific needs of the quantum algorithm, considering factors such as quantum circuit structure, layout of qubits, resource availability, latency, and the hardware characteristics of the QPUs. Hybrid approaches that combine both primitives are often used to minimize communication and optimize performance [113], [114].

In any case, both TeleData and TeleGate rely on E2E entanglement to operate effectively. For example, in a quantum data center with a DQC architecture, as the demand for executing remote gates increases, an overlay network for E2E entanglement distribution becomes essential. To facilitate this, each QPU must allocate at least one communication qubit to generate direct-link entangled pairs. Networking protocols then utilize these pairs to establish E2E entanglement between end nodes, serving as a fundamental communication primitive.

Existing research on entanglement distribution often assumed a single E2E entanglement request at any given time, requiring only one path between two nodes [115], [116], [117]. However, quantum data centers typically handle multiple simultaneous requests, as quantum circuits often involve numerous remote gates, each dependent on E2E entanglement. Some gates can be executed in an arbitrary order, while others must be performed sequentially, as they depend on the outcomes of preceding gates. Additionally, certain gates are restricted to specific QPUs, whereas others can be executed on any QPU. These scenarios necessitate managing the allocation of entangled qubits between neighboring nodes, leading to variations in link capacities and emphasizing the importance of fair and efficient resource allocation [118], [119], [120].

1.3.5 Compilation Pipeline for Distributed Quantum Circuits

The components discussed in Subsections 1.3.2, 1.3.3, and 1.3.4 form integral parts of the quantum circuit compilation pipeline necessary for DQC. This subsection presents the overall compilation workflow, highlighting the key transformations involved in preparing high-level quantum algorithms for execution across a network of interconnected QPUs.

Unlike traditional monolithic systems, DQC demands additional layers of optimization and coordination to account for inter-QPU communication, entanglement distribution, and remote gate

execution. By incorporating circuit partitioning, hardware-aware gate decomposition, qubit mapping, and inter/intra-QPU routing, this workflow ensures that circuits are efficiently adapted to the constraints and capabilities of DQC architectures. The final scheduling and backend-specific translation steps prepare the circuit for execution across multiple QPUs, enabling scalable computation. As quantum hardware and network infrastructure continue to mature, such structured compilation processes will play a vital role in harnessing the full potential of DQC.

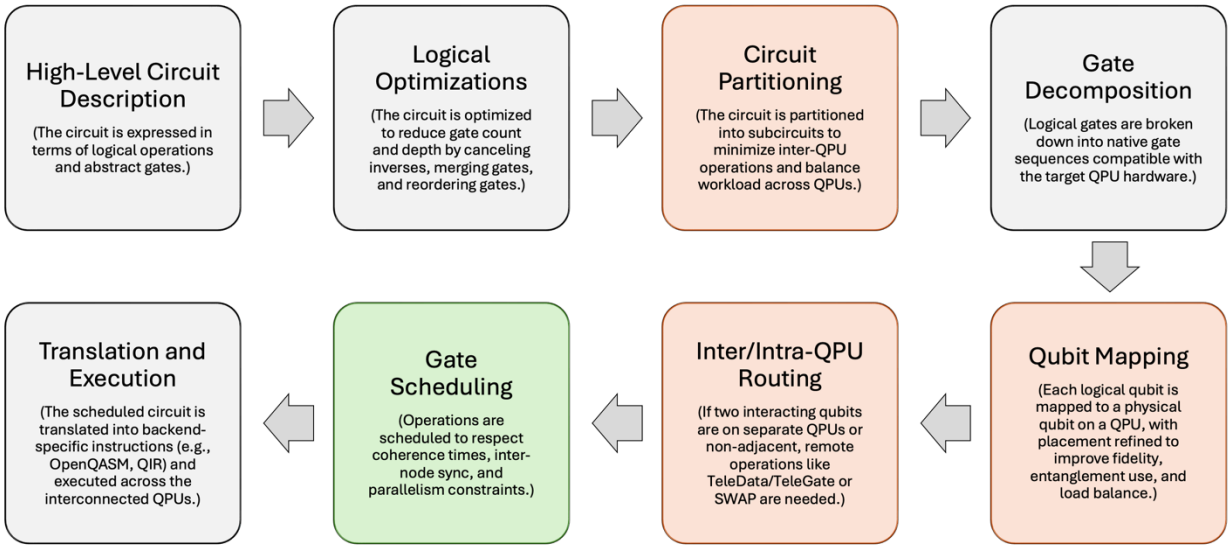


Figure 1.8. *Compilation Pipeline for Distributed Quantum Circuits.*

Figure 1.8 presents the overall compilation workflow, with colored boxes highlighting steps tailored for DQC. The orange boxes correspond to components discussed in Subsections 1.3.2, 1.3.3, and 1.3.4, respectively, while the green box represents a step that will be covered in detail in Chapter 4. The process begins with a high-level circuit description, which is then simplified through logical optimizations to reduce gate count and circuit depth. The optimized circuit is partitioned into subcircuits assigned to different QPUs to minimize inter-node communication and balance the computational load. Following partitioning, gate decomposition translates high-level gates into native gate sequences compatible with the target hardware. Each logical qubit is then

mapped to a physical qubit on a specific QPU, with placement refined to improve fidelity and entanglement efficiency. Once qubit locations are established, inter- and intra-QPU routing mechanisms, such as SWAP insertion or teleportation-based operations (TeleData/TeleGate), are applied to support remote interactions. The circuit is then scheduled with consideration for coherence times, synchronization constraints, and execution parallelism. Finally, the scheduled instructions are translated into backend-specific formats and executed across the distributed quantum hardware.

1.4 Quantum Internet Security

As quantum networks evolve toward large-scale deployment, security becomes a foundational concern, particularly due to the fundamental differences between classical and quantum communication paradigms. The no-cloning theorem and the collapse of quantum states upon measurement introduce distinct security challenges as well as novel opportunities for protection in quantum networks. The quantum internet, envisioned as a global-scale entanglement-based or qubit-based communication infrastructure, must be designed to guarantee the CIA of quantum data and protocols across all layers of the network stack [15], [27], [121], [122]. While the quantum internet offers inherently strong security features for communication, its underlying architecture, particularly when integrated with classical networks or applied to cryptographic tasks like QKD, remains vulnerable and demands careful consideration.

For example, at the physical layer, quantum channels are vulnerable to environmental noise, photon loss, and decoherence, which can be exploited by attackers using strategies like Photon Number Splitting (PNS) [123] or Trojan-horse attacks [124]. Countermeasures include decoy-state protocols, signal filtering , and hardware-level defenses [125]. The link and network layers must

address challenges like malicious or compromised nodes (e.g., untrusted repeaters) and Denial-of-Service (DoS) attacks that can degrade entanglement distribution. Trusted node models or advanced techniques like device-independent QKD and quantum authentication protocols are crucial for mitigating these threats [126]. In routing protocols, adversaries could misreport entanglement availability or manipulate measurement outcomes to reroute traffic inefficiently or disrupt long-distance entanglement generation. Protocols must be designed to incorporate redundancy, verification, and fault isolation. The control and management plane should support secure coordination and robust scheduling to defend against injection or manipulation of control messages. On the application layer, the quantum internet enables cryptographic applications such as QKD and quantum digital signatures but also introduces new attack vectors that must be addressed in a scalable and hardware-aware fashion. For example, scalable multi-party QKD, distributed consensus using entanglement, and quantum blockchains require both quantum and classical authentication mechanisms [127], [128].

A comprehensive security strategy for the quantum internet must therefore be layered, adaptive, and hardware-aware, integrating both quantum-native primitives and classical protections [129]. Ongoing research explores models for quantum intrusion detection, quantum firewalls, and cross-layer threat mitigation frameworks that account for both technological constraints and adversarial behavior [130], [131]. As quantum hardware and networking protocols continue to mature, security must remain a core design principle, not an afterthought, to ensure trustworthiness, resilience, and global adoption of quantum communication technologies.

1.5 Related Work

Scaling quantum computing has been a long-standing challenge, supported by a substantial body of research over the years. This section reviews existing research related to quantum networks, DQC, and quantum network security.

1.5.1 Entanglement Routing

Entanglement routing is a fundamental component of quantum networks, tasked with establishing high-fidelity entangled pairs between distant nodes to support applications such as quantum teleportation, sensing, and QKD. Unlike classical routing, which forwards digital packets across deterministic links, entanglement routing must contend with the inherently probabilistic nature of quantum link generation, limited coherence times, and non-reusable resources such as entangled pairs. As a result, designing routing protocols for quantum networks requires a fundamentally different approach, one that accounts for temporal fluctuations, resource constraints, and the need for coordination between quantum and classical control planes.

Numerous strategies have been proposed to improve the efficiency and scalability of entanglement routing. Many of these works aim to maximize the availability and throughput of E2E entangled pairs while minimizing latency and entanglement consumption. For example, works such as [15], [21], [116], and [52] provide foundational models for quantum network architectures, establishing key components such as quantum repeaters, BSM modules, and quantum memory buffers. These studies also introduce and analyze critical metrics for evaluating routing performance, including entanglement generation success probability, link fidelity, and expected number of successful E2E paths under varying network topologies and hardware conditions.

Additionally, several studies have explored dynamic and adaptive routing protocols designed to establish E2E entanglement under realistic and often stringent resource constraints [115], [132], [133]. Unlike static approaches that rely on predetermined routes or assume stable link conditions, these dynamic protocols adapt to the network state, accounting for variations in link availability, quantum memory occupancy, entanglement success rates, and the fidelity of quantum channels. This adaptability is especially important in quantum networks, where entanglement links are probabilistic, short-lived, and highly sensitive to environmental noise and hardware imperfections.

To enhance routing efficiency, some of these methods incorporate techniques from classical networking, such as multipath routing, graph-based heuristics, and traffic engineering mechanisms [117], [134], [135], [136]. These classical principles are modified and extended to fit the quantum context. For instance, multipath routing strategies are tailored to account for the probabilistic success of entanglement generation across parallel paths, enabling redundancy to improve robustness and throughput. Graph-based routing algorithms are adapted to consider metrics like expected fidelity, quantum memory lifetime, and available entangled link quality, rather than traditional bandwidth or hop count.

Despite these advancements, most existing techniques rely on synchronization-based routing approaches, which require precise timing and global coordination among quantum nodes. These methods depend on the propagation of up-to-date network status across the entire system before routing decisions can be made. However, this approach is impractical for entanglement routing, as the short coherence time of direct-link entanglements limits how long these links remain usable. By the time network status is fully disseminated, many entanglement links may have already decohered, rendering the routing information outdated and the protocol ineffective. Chapters 2 and 3 of this dissertation addresses these gaps.

1.5.2 Resource Management in DQC

Many existing studies simplify the networking problem by assuming only a single user request is present at any given time, requiring just one entanglement path. However, in real-world scenarios, particularly in DQC environments, multiple users or applications often request quantum resources simultaneously, creating the need for concurrent path establishment and more sophisticated resource coordination.

Efficient resource management is thus crucial for supporting distributed quantum tasks such as entanglement generation, teleportation, and remote gate execution, particularly in cases involving simultaneous user requests. Works such as [137], [138], [139], and [140] have developed optimization models to allocate quantum resources effectively, considering factors like link capacity and fidelity requirements. Additionally, reinforcement learning has been explored in studies like [141] and [142] to address the dynamic nature of quantum resource demands. Despite these advances, practical implementations are constrained by the inherently short-lived nature of quantum states and the necessity for precise synchronization across nodes.

DQC relies on partitioning large quantum circuits across multiple QPUs to overcome the qubit limitations of individual processors. Early work on DQC, such as [143], [144], and [19], introduced foundational approaches for quantum circuit partitioning and remote gate execution using teleportation and entanglement swapping. More recent studies, including [145] and [146], have addressed optimizing communication overhead by developing strategies that reduce the number of remote gates and enhance the fidelity of shared entanglement. While these efforts have laid a strong foundation, challenges remain in ensuring efficient coordination between QPUs and minimizing communication delays in large-scale quantum systems.

While significant progress has been made in these areas, to the best of our knowledge, no existing resource management framework specifically targets DQC by abstractly modeling the components of a partitioned quantum circuit executed in a distributed environment. Furthermore, there is limited research focused on developing effective and efficient algorithms for resource management in DQC or quantum networks. Given the current scale of DQC architectures, simple algorithms could suffice in many cases. However, most existing works emphasize sophisticated methods such as reinforcement learning or optimization solvers, which may compromise practicality due to their complexity, offering only marginal performance improvements. Additionally, few studies address the trade-offs between execution time and fidelity in resource-constrained environments. Chapter 4 of this dissertation addresses these gaps.

1.5.3 Quantum Internet Security

Security has been a central theme in the development of quantum communication systems, with foundational work focusing on the information-theoretic guarantees provided by protocols like QKD. Since the introduction of the BB84 protocol by Bennett and Brassard in 1984 [49] and the E91 protocol by Ekert [3], extensive efforts have explored both the theoretical and experimental dimensions of QKD, including decoy-state methods [125], measurement-device-independent QKD (MDI-QKD) [147], and continuous-variable QKD [148]. These protocols provide strong confidentiality guarantees under well-defined models but often assume trusted nodes and idealized devices, which are difficult to maintain in real-world network deployments.

As research shifts toward quantum networking and the quantum internet, security challenges expand beyond point-to-point QKD. Several studies have examined vulnerabilities arising from physical-layer attacks, including photon-number splitting, Trojan-horse attacks, and detector blinding, which exploit imperfections in quantum hardware [123], [124], [149]. Proposed

countermeasures include decoy-state techniques, optical isolators, and active monitoring of channel characteristics [125], [150], [151]. At the network layer, attention has turned to untrusted quantum repeaters, which may disrupt entanglement generation or manipulate entanglement swapping operations [129]. Work on entanglement verification protocols and authentication of measurement outcomes addresses these concerns [152], [153], though scalable implementations remain challenging.

Emerging concepts such as quantum intrusion detection systems, and post-quantum resilient protocols represent promising directions for future research [130], [131], [154]. However, many of these systems remain in early conceptual or simulation-based stages, and substantial work is still needed to address performance trade-offs, real-time threat detection, and integration with heterogeneous network topologies.

While foundational research in quantum cryptography has laid a strong groundwork, the field of quantum internet security remains in its early stages. Existing approaches often focus on isolated layers or idealized assumptions, and there is a growing need for practical, E2E security architectures that account for quantum hardware limitations, interdependence with classical infrastructure, and dynamic network behavior. However, cross-layer security frameworks that integrate classical and quantum mechanisms to ensure CIA across hybrid architectures is lacking, as well as the potential cryptographic applications in the quantum era. Chapters 5 and 6 of this dissertation addresses these gaps.

1.6 Goals and Contributions

With the overarching research objective of improving the scalability, security, performance, and practical deployment of both quantum and classical systems and networks, this section outlines the specific goals and key contributions of this dissertation.

1.6.1 Goal 1: Practical and Efficient Entanglement Routing

Many existing routing techniques rely on global synchronization and network-wide status updates, which are impractical for entanglement routing due to short coherence times. By the time routing information propagates, many entangled links may have already decohered, making the decisions outdated and ineffective. Goal 1 seeks to overcome these limitations by developing a practical and efficient entanglement routing scheme.

Contribution 1.1. *Asynchronous entanglement routing based on local knowledge of link states*. We propose Asynchronous Entanglement Routing (AER) to eliminate the need for synchronized operations and increase utility, significantly improving E2E entanglement efficiency [117], [155], [156]. AER uses only local neighbor link states and structures like Destination-Oriented Directed Acyclic Graphs (DODAGs) or spanning trees to route entanglement in a distributed manner. By avoiding network-wide updates, AER enhances practicality and achieves higher rates than synchronous methods, making it a promising solution for future quantum networks.

Contribution 1.2. *Multi-tree entanglement routing based on AER*. We extend AER to a multi-tree scheme for larger-scale quantum networks by connecting multiple DODAGs in a distributed, asynchronous manner using only local link knowledge [134]. Tests on generated and realistic topologies show that this approach outperforms single-tree and synchronous routing in entanglement rate and scalability.

1.6.2 Goal 2: Efficient and Scalable Scheduling for DQC

Many existing studies assume a single user request at a time, overlooking the need for concurrent path setup and coordination in real-world DQC scenarios. Even when simultaneous requests are considered, most focus on complex methods like reinforcement learning or optimization solvers, which add overhead with limited performance gains. Also, few consider execution time vs. fidelity trade-offs in resource-limited settings. Goal 2 addresses these gaps by developing an efficient and scalable scheduling scheme for DQC.

Contribution 2. *Efficient Scheduling for DQC in Quantum Data Centers*. We propose a resource allocation framework for quantum data centers that maximizes fidelity-guaranteed throughput while preserving dependencies [157]. Using a MILP model for benchmarking, we develop efficient heuristics that closely match solver results, showing strong practicality for managing remote gates and enhancing performance.

1.6.3 Goal 3: Secure and Resilient Quantum Internet

Existing approaches often target isolated layers or idealized conditions, highlighting the need for practical E2E security architectures that consider hardware limits, classical integration, and network dynamics. Cross-layer frameworks ensuring CIA in hybrid systems, along with support for quantum-era cryptographic applications, remain underdeveloped. Goal 3 addresses these gaps by developing a secure and resilient quantum network alongside cryptographic applications such as quantum blockchains.

Contribution 3.1. *Layer-wise Security Framework for Quantum Networks*. We propose a security framework by analyzing vulnerabilities and mitigation strategies across the quantum internet stack (i.e., physical, link, network, and application layers), assessing attack severity and countermeasure

effectiveness using both classical and quantum methods [129]. Findings underscore the evolving nature of security threats and the need for adaptive defenses.

Contribution 3.2. *Theoretical design for Quantum Blockchains*. We explore quantum-era cryptographic applications by analyzing quantum blockchains for decentralized identity, proposing a conceptual framework and evaluating its feasibility, effectiveness, and key components [158].

The remainder of this dissertation is structured as follows. Chapters 2 and 3 presents our contributions related to Goal 1 (c.f., quantum networks). Chapter 4 explores the work addressing Goal 2 (c.f., DQC), while Chapters 5 and 6 details the contributions toward Goal 3 (c.f., quantum network security). Finally, Chapter 7 summarizes the findings and outlines potential directions for future research.

Chapter 2: Asynchronous Quantum Routing

In this chapter, we propose a set of asynchronous routing protocols, AER, for quantum networks by maintaining a dynamic topology in a distributed manner, which has been extensively studied in classical routing for lossy networks, such as using a DODAG or a spanning tree. The protocols update the entanglement-link topology asynchronously and identify practical paths based on local knowledge of network status. Simulation results indicate that AER achieves a larger upper bound and significantly higher rates than existing synchronous approaches, and the rates increase with coherence time, suggesting that it will have a much more profound impact as technology advances.

2.1 Introduction

Quantum networks [1], [15], [30] offer many possibilities for applications that go beyond what classical networks can accomplish. Among these are sensing [4], secure communication [2], QKD [159], clock synchronization [160], and DQC [20], [161]. Quantum repeaters have been developed using various techniques to facilitate long-distance communication for these applications. These methods can be broadly categorized into two groups: the first involves transmitting encoded quantum information [22], [66] and applying quantum error correction [162], similar to classical repeaters. The second, hereafter referred to as the quantum-native repeaters (or in some literature, first-generation repeaters [67]), distributes long-distance entanglement [21] through entanglement swapping [163] (Figure 2.1), allowing end-users to carry out desired operations such as QKD or secure communications. While a direct application of classical networking technologies to quantum networks by transmitting encoded quantum states may seem intuitive, the quantum-native repeater provides a more immediate solution to creating a network that utilizes E2E entanglement,

as the former approaches necessitate a considerably impractical quantity of qubits for error correction purposes [10], [164].

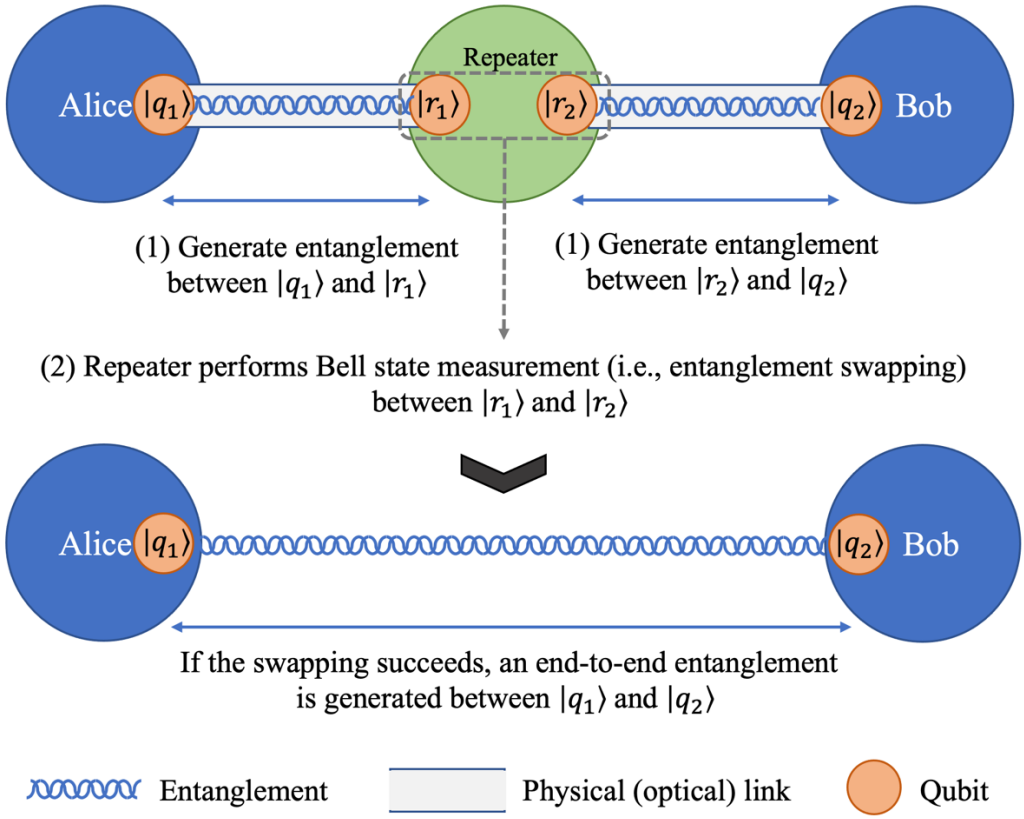


Figure 2.1. *Single-hop Entanglement Generation via Entanglement Swapping.*

As shown in Figure 2.1, Alice and Bob intend to establish an E2E entanglement through a repeater positioned in between. The process involves two steps: (1) both parties create an entangled state with the repeater, resulting in two pairs of entangled qubits: Alice's $|q_1\rangle$ paired with the repeater's $|r_1\rangle$ and Bob's $|q_2\rangle$ paired with the repeater's $|r_2\rangle$; (2) the repeater performs entanglement swapping, which essentially performs BSM on $|r_1\rangle$ and $|r_2\rangle$ (see Subsection 2.2.1) assisted by classical communication, producing an E2E entanglement between Alice's $|q_1\rangle$ and Bob's $|q_2\rangle$.

The development of direct-link entanglement generation has brought quantum networks closer to realization, with recent experiments demonstrating deterministic entanglement delivery between

solid-state memories located in different physical locations [52], facilitated by the use of telecommunication wavelengths to extend the distance between the locations [165], [166]. However, the rate of direct-link entanglement generation still decreases significantly over long distances due to photon loss and quantum state and operation instability resulting from unavoidable interactions with the environment. Quantum repeaters based on entanglement swapping are thus necessary to extend the entanglement generation (distribution) distance. However, they may not always be capable of achieving the desired fidelity levels required for certain applications [74], [167]. Entanglement purification (see Subsection 2.2.2) can improve fidelity, but it has a trade-off. It reduces the number of entangled pairs shared between neighboring nodes on the network and requires a substantial amount of classical communication overhead [89]. Entanglement swapping and purification form a quantum-native repeater scheme analogous to the link layer of classical networks (see Appendix A).

On top of the quantum-native repeater scheme, a network layer can be established to ensure an effective architecture [168], [169] for quantum networks, thereby enabling the full network capabilities. The quantum network layer plays a crucial role in determining the routing principles, i.e., quantum-native routing, that facilitate requests for generating E2E entanglements between remote stations [116], [170], [171]. Various metrics have been suggested to direct the routing process [115], [172], and the entanglement rate — the number of entanglements generated per unit time — is a prevalent metric akin to throughput in classical networks, with entanglements (as the communication resources) replacing bits. In essence, designing a quantum-native routing protocol aims to determine the optimal path while maximizing the entanglement rate between any two nodes under specific constraints, such as operation probability, coherence time, and link capacity. It should be noted that a classical network is assumed to be present in the quantum network,

responsible for performing routing computations and broadcasting routing information. In other words, each node in the quantum network must be able to communicate with one another via a classical network, c.f., LOCC. This assumption can be safely made given the expectation of a hybrid quantum-classical network and is a common practice in quantum routing techniques [115], [116], [170], [171], [172]. After the paths have been established with the aid of LOCC, the intermediate nodes (repeaters) carry out entanglement swapping and purification. The resulting E2E entanglements connecting the endpoints (end users) are then used as the communication resources for any desired operations. The nature of E2E entanglement resembles connection-oriented circuit-switching, in contrast to the connection-less packet switching commonly used in classical networks, which was also proposed for quantum networks [173]. To avoid confusion, the term "quantum-native" is explicitly used to denote that the technique applies only to quantum networks (to distinguish from the repeater approaches that relay qubits) and is not independent of classical networks.

The routing process in existing approaches [115], [116], [170], [171] is divided into two phases, namely external and internal, each assigned a specific time slot. During the external phase, pairs of network nodes with a direct connection try to establish entanglement using a physical link, resulting in an *instant topology* (Figure 2.2(a) and 2.2(b)) consisting of direct-link entanglements. In the internal phase, assuming that nodes are only aware of the direct-link entanglement generation results of their neighbors (see Subsection 2.2.3), each repeater node swaps entanglement blindly to reach an E2E entanglement between the two endpoints that request to communicate (Figure 2.2(c) and Figure 2.2(d)). If the E2E entanglement is not achieved, the two phases are repeated. It is essential to synchronize both phases to ensure that entanglement "links" remain available for the internal phase. However, this synchronization and blind attempts consume

all entanglements in each time slot, leading to a low entanglement rate. Furthermore, there are no obvious solutions for connecting networks of networks in these existing approaches.

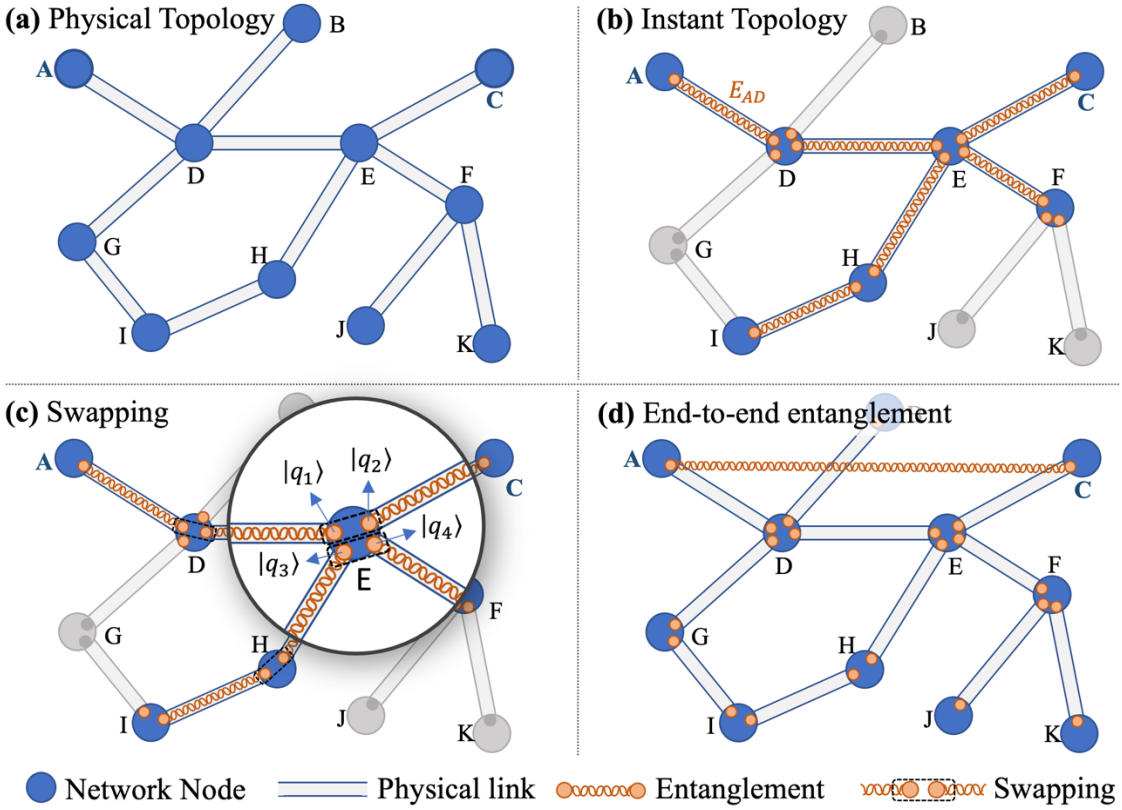


Figure 2.2. *Topologies in Different Phases and E2E Entanglement Generation.*

As shown in Figure 2.2, Nodes A and C in the network intend to establish an E2E entanglement. In Figure 2.2(a), the network consists of nodes connected by optical communication links, forming the *physical topology*. In the external phase shown in Figure 2.2(b), nodes connected by the same optical link attempt to generate direct-link entanglement, resulting in an *instant topology* of entanglement links (the orange helical links). For example, Nodes A and D are connected by the entanglement link E_{AD}. In Figure 2.2(c), all repeaters involved in two entangled pairs perform entanglement swapping on the qubits closer to the source or destination. Node E, for instance, has four qubits, with each qubit entangled with one of its four neighboring nodes (E_{DE}, E_{HE}, E_{EF}, and E_{EC}). It performs swapping on $|q_1\rangle$ and $|q_2\rangle$ since they are connected to the nodes closer to A and

C, respectively. In Figure 2.2(d), by consuming all entanglements in the instant topology, A and C successfully obtain an E2E entanglement. Since each node is only aware of itself and its neighbor's entanglement-link statuses, Node E does not know if the path A-D-E-C would succeed. It also performs swapping on $|q_3\rangle$ and $|q_4\rangle$ to increase the success rate of achieving an E2E entanglement.

In this chapter, we propose a set of asynchronous routing protocols that can find the path between any two endpoints in a network with a much higher entanglement rate than existing approaches. Also, the entanglement rate increases with coherence time, and there are gateways in the network to connect networks of networks. It removes the need for synchronized executions by iterating an instant topology consisting of entanglement links. The instant topology can be in the form of a graph that can be maintained in a distributed manner, such as a destination-oriented directed acyclic graph (DODAG) or a spanning tree. Such techniques have been well-studied for classical lossy networks, which share the similarity of losing a connection (in a quantum context, an entanglement in the instant topology) with a probability depending on the quality of the physical link. Our theoretical analyses and simulations demonstrate that our asynchronous protocols achieve a larger upper bound and significantly higher entanglement rate compared to existing synchronous approaches. The entanglement rate increases as coherence time increases. Additionally, a tree-like instant topology such as DODAG allows for the connection of networks of networks through the root nodes. These findings suggest that our proposed protocols will have a significant impact on the development of a quantum internet as technology advances (as coherence time increases and networks expand in scale).

The rest of the chapter is organized as follows. First, we define the problem statement of this research by introducing the routing metric and the settings of the existing routing scheme we aim

to optimize. Then, we propose the framework and two example algorithms for the asynchronous routing protocols that maintain an instant topology in a distributed fashion. Next, we evaluate the routing performance from various perspectives and provide a detailed explanation of a sample routing result with DODAG. Subsequently, we analyze the simulation results to examine the effectiveness of our scheme under different conditions and establish its superiority. Finally, we conclude the chapter by discussing the limitations, the extent of usability, and potential areas for future research and expansion.

2.2 Preliminaries

This section provides an overview of key concepts and methodologies relevant to our research. It offers essential background to ensure readers grasp the subsequent discussions and the context of our research question.

2.2.1 Entanglement Swapping

Entanglement swapping, in essence, is to perform a BSM on two pairs of entangled qubits. In the circuit depicted in Figure 2.3, three parties are involved: Alice, Bob, and a repeater. Alice possesses a qubit $|q_0\rangle$, the repeater has qubits $|q_1\rangle$ and $|q_2\rangle$, and Bob has a qubit $|q_3\rangle$. Before t_1, operations (i.e., Hardamard and CNOT gates) are carried out to produce two pairs of entangled qubits: one between Alice and the repeater, $|q_0q_1\rangle$, and another between the repeater and Bob, $|q_2q_3\rangle$. Following this, the repeater executes a BSM, which is used initially to identify the specific Bell state that an entangled pair is in. After that, at t_2, the two entangled pairs $|q_0q_1\rangle$ and $|q_2q_3\rangle$ are swapped to form a new, single pair of entangled qubits, $|q_0q_3\rangle$, which connects Alice and Bob with an entangled pair (i.e., an entanglement link).

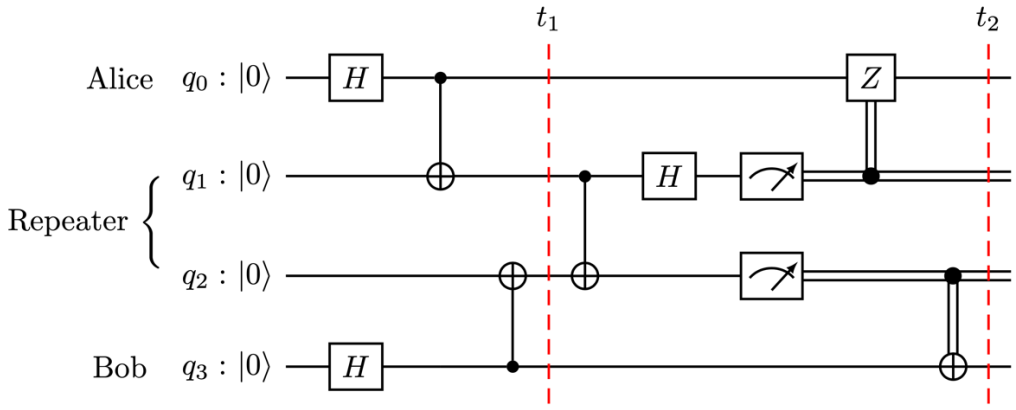


Figure 2.3. *The Circuit of Entanglement Swapping.*

2.2.2 Entanglement Purification

At time t_1, Figure 2.4 demonstrates that Alice and Bob possess two entangled pairs, namely $|q_1q_2\rangle$ and $|q_0q_3\rangle$, indicating a link capacity of two between them. This implies that two raw entanglements with moderate fidelity will be utilized to produce a single entanglement with higher fidelity. Alice owns $|q_0\rangle$ and $|q_1\rangle$, while Bob owns $|q_2\rangle$ and $|q_3\rangle$. Both $|q_1q_2\rangle$ and $|q_0q_3\rangle$ are considered low-quality (raw entanglements). Successful purification is determined by coincidence measurements in the Z basis. After the purification process, one of the entangled pairs is consumed while the other becomes a higher-quality entanglement. In this example, $|q_0q_3\rangle$ is consumed, and $|q_1q_2\rangle$ is preserved.

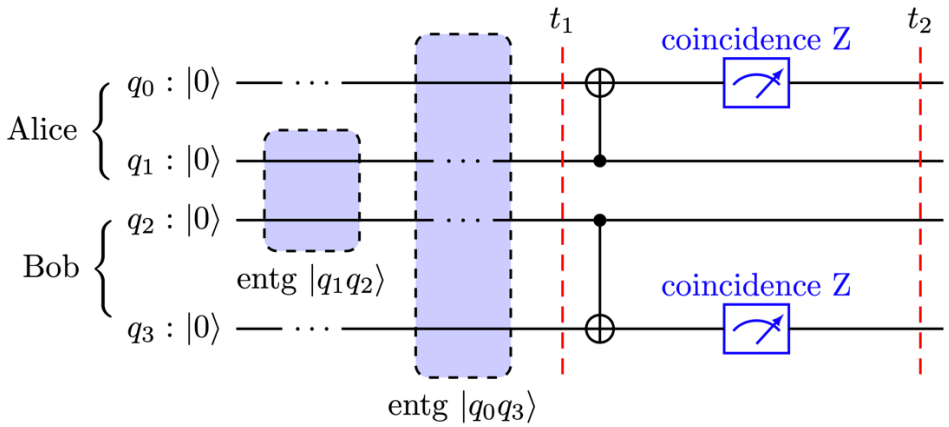


Figure 2.4. *An Example Purification Circuit with Link Capacity Two.*

2.2.3 Global and Local Knowledge of Network Status

For a quantum routing protocol to be effective, it must be capable of identifying the optimal path as and when required. If a previously selected path becomes unavailable, the proposal must be able to identify an alternative path from the current location to the destination. Since the global knowledge of the entire network's entanglement links is often not up to date, the protocol must identify the best adjacent node based on its local knowledge of the entanglement links connected to it. Although the shortest path in an instant topology can be easily identified if a global view is available, it is unrealistic to assume every node has the latest version of it (e.g., to assume every node instantly knows the instant topology created by nodes A, D, E, H, I, F, C in Figure 2.2. The status of all direct-link entanglements can only be propagated through classical networks, which is relatively slow. In large networks, the instant topology may have already been altered by the time the global view has been broadcast to every network node. As a result, routing should be constrained to ensure that each node is only aware of the entanglement link states of its neighbors (e.g., node D only knows the direct-link entanglement generation results of links D-A, D-B, D-G, and D-E. Specifically, D-A and D-E have been successful, while the other two have failed).

2.2.4 Finding Disjoint Paths

In a network, disjoint paths refer to two or more paths between a pair of nodes (i.e., source and destination) that do not share any common node or link/edge except for the source and destination. Here, we consider link-disjoint paths where two paths do not have any link or edge in common.

Finding disjoint paths is necessary in reality due to the current low link capacity in quantum networks. Assuming global knowledge of the instant topology, finding even two disjoint shortest paths in many scenarios remains an NP-complete problem, such as the min-min (i.e., to minimize

the length of the shorter path of the two disjoint paths), min-max (i.e., to minimize the length of the longer path of the two disjoint paths), and min-sum (i.e., to minimize the sum of the length of the two disjoint paths) problems. Besides single-cost min-sum, others are NP-complete. In addition, it is NP-hard to obtain a ρ-approximation to the optimal solution for any $\rho > 1$. More details can be found in Reference [174]. This increases the time needed for classical operations to find the shortest paths. Calculating paths and propagating the global view of the instant topology within the entanglement coherence time is unrealistic. However, this problem will be eased in the future if there is a high link capacity.

2.2.5 Link Capacity and Desired Fidelity

The maximum number of entanglements that can be generated between two parties is referred to as link capacity. As illustrated in Figure 2.5, Alice and Bob each have two qubits, which means they can generate two entangled pairs, leading to resource/qubit allocation and scheduling issues. For more information, please refer to Reference [170]. However, if there is redundant link capacity, a link in the instant topology can last longer since the other pairs of qubits (besides the one being used) will continually attempt entanglement generation, providing backup direct-link entanglements. Redundant entanglements in one edge can also be utilized to generate multiple paths and support simultaneous connection requests. However, having four qubits does not necessarily mean we can achieve the maximum of two entanglements since various entanglement-based applications, such as QKD, require a minimum fidelity level. Some entanglements must be used up to attain the desired fidelity, as explained in Subsection 2.2.2. Consequently, the actual link capacity should be $C/E(f)$. Assuming homogeneous link and node quality, this is a constant factor applied to the entanglement rate. In the main text, without loss of generality, we set the link capacity $C = 1$ and the number of entanglements for purification $E(f) = 1$ since they are

constant as average numbers. This simplifies the routing problem and allows us to focus on the entanglement rate change by maintaining a distributed graph.

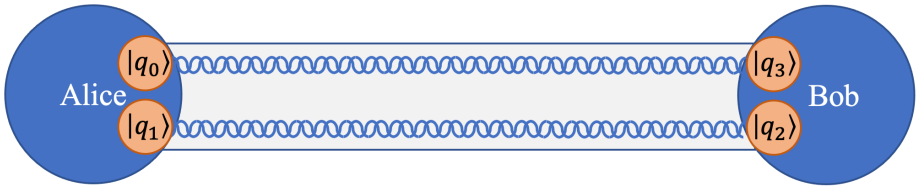


Figure 2.5. *A Link between Alice and Bob with Capacity Two.*

2.3 The Entanglement Routing Problem

Quantum networks have several constraints that do not apply to classical networks. One such constraint is the probabilistic nature of entanglement generation and swapping performed by repeaters. Without loss of generality, we consider the generation of the maximally entangled states (Bell states) between adjacent nodes and the operation of BSM at repeater nodes to swap entanglement. Each instance of generation or swapping has a chance of succeeding or failing, leading to probabilistic and unpredictable routing paths. As pathfinding takes place and direct-link entanglements are generated, the instant topology changes constantly, emphasizing the need for an adaptive routing approach to creating an E2E entanglement. Finally, once an E2E entanglement is consumed, the network must establish a new one for a new connection request. In short, E2E entanglements allow the creation of dedicated unicast channels between node pairs, regardless of their position in the network. As mentioned, this characteristic of entanglement resembles connection-oriented circuit-switching rather than the connection-less packet switching used in classical networking. Here, we present a definition of the quantum-native routing problem.

Let us consider a graph $G(V, E)$ that represents the physical topology of the quantum network (Figure 2.2(a)). In this graph, each node $v \in V$ represents a quantum-native repeater, while each

edge $e \in E$ denotes an optical link that connects two repeater nodes. The instant topology (Figure 2.2(b)) is denoted as $G'(V', E')$, which is a subgraph of G. E' is the set of direct-link entanglements, and V' are the nodes that are connected by the entanglements in E'. Additionally, each edge (link) e can support multiple entanglement generations with multiple qubit pairs between adjacent nodes, where the link capacity $C(e)$ of link e is defined as the maximum number of entangled pairs generated between adjacent nodes connected by e. Each node contains a finite number of qubits (depicted as orange dots in Figure 2.2), denoted as node capacity $C(v) \geq \sum_{e \in N(v)} C(e)$ where $N(v)$ refers to the set of links connecting node v. To achieve a desired fidelity f, we need $E(f)$ entanglements for purification, and each entanglement has an average coherence time, T_{co}. For simplicity, we assume that the link capacity C and purification number $E(f)$ are both equal to 1. This implies that, as depicted in Figure 2.2, only one qubit is present on one side of the physical link. Nonetheless, varying link capacities and purification numbers will not impact the evaluation of the routing scheme as they simply introduce a constant multiplier to the entanglement rate. The entanglement rate will increase linearly relative to the actual link capacity, as detailed in Subsection 2.2.5. Given our assumption that $C/E(f) = 1$, we can omit the $C/E(f)$ multiplier from each entanglement rate equation.

In addition, for each entanglement generation between a direct link e, there is a probability $p(e)$ that the generation will succeed, while each entanglement swapping at a repeater v has a probability $q(v)$ of success. For convenience, we assume homogeneous entanglement generation probability, denoted by p, and entanglement swapping probability, denoted by q, on all network links and nodes. Finally, the entanglement rate, denoted by ξ, is the performance metric indicating the number of E2E entanglements (Bell states) that can be generated in a unit time T. T must satisfy $T \leq T_{co}$. Existing approaches usually consider the time required for both internal and external

phases (as discussed in the Section 2.1) to be one unit time and assume it to be longer than the coherence time. To facilitate comparison, we set the unit time as $T = T_{co}/n$, where $n \in \mathbb{Z}$ and $n \geq 1$.

With these preliminaries, the problem of quantum-native routing is to design an effective algorithm that maximizes the entanglement rate under the constraint that nodes only possess local knowledge of adjacent links in the instant topology (For example, Node E in Figure 2.2(b) only knows the direct-link generation results of E_{ED}, E_{EH}, E_{EF}, and E_{EC}). Note that it is unrealistic to assume that a global view of the instant topology is known, which may allow an algorithm to determine the shortest paths in the instant topology. In existing protocols such as [116], [170] that depend on global knowledge of the instant topology, entanglements need to possess longer coherence times as the network size (i.e., the number of network nodes) expands. This is because direct-link entanglement information takes a relatively long time to traverse the entire network. Consequently, in a large-scale quantum network, direct-link entanglements may decay by the time all direct-link generation results are collected. Thus, a distributed approach is necessary to find optimal paths with local knowledge of the instant topology.

Nonetheless, even if we assume to have a global view of the instant topology, it is still NP-complete to find multiple disjoint shortest paths (for robustness) in many scenarios (see Subsection 2.2.4), given the restricted resources available at each link [175]. However, many existing approaches assume that the global view of the instant topology is known, such as [170], [171], [115], [31]. A recent work by Pant et al. [116] proposes a blind search routing strategy with local knowledge of the instant topology, which, however, requires synchronized phases and consumes all direct-link entanglements in each attempt, as shown in Figure 2.2(c) and 2.2(d), resulting in a low entanglement rate. To ensure efficiency, entanglement swapping within a repeater should only

occur when a path that includes that node is selected. Our approach, therefore, involves maintaining a distributed graph and reserving entanglements not used in the current paths for future use.

2.4 Asynchronous Routing Scheme

In essence, our method of asynchronous routing functions on an event-driven basis. It establishes and maintains the instant topology as a distributed graph (such as DODAG or spanning tree) and reacts to connection requests (events) by following the path determined in the distributed graph, such as tracing the path through the most current DODAG's root. The crux of the matter is the former service, which outlines how to preserve the instant topology as a distributed graph that determines a route for any given pair of nodes. In particular, we illustrate the protocol using two sample graphs: DODAG and spanning tree. Protocol 1 in Figure 2.6 presents the algorithmic steps of the routing processes, including quantum operations (QO), such as entanglement generation and swapping, and classical operations (CO), such as event listening and request handling. Moreover, the maintenance of the two example graphs (DODAG and spanning tree) is illustrated later in Protocol 1.1 in Figure 2.8 and Protocol 1.2 in Figure 2.10, which are replaceable by any distributed routing algorithms.

Protocol 1 is executed on each network node. To join the network, any node has to only abide by the protocol without requiring synchronization of operations and communications across all nodes. Lines 1 and 2 of Protocol 1 define the interfaces for executing classical operations (i.e., CO) and conducting quantum operations (i.e., QO). Line 3 assigns the identifier of the current node running the protocol instance to N_c.

Algorithm 1 Asynchronous routing scheme

(N_{prev}, N_{next}: the previous and next node in the path.)
(N_{dest}: destination.)

▷ **Resource**: System Interface and Variable
1: $CO \leftarrow$ A set of classical operations
2: $QO \leftarrow$ A set of quantum operations
3: $N_c \leftarrow$ The current node that is running a protocol instance

▷ **On-demand Service:** Topology Maintenance and Listener
4: GRAPHUPDATE(QO, CO, N_c) ▷ Refer to Protocol 1.1 or 1.2
5: CO.LISTEN('connection_request', N_{prev}, N_{dest})

▷ **Request Handler:** Path Navigation and Swapping
6: **function** NAVIGATE(N_{prev}, N_{dest})
7: $N_{next} \leftarrow$ NEXTHOPDETERMINATION(N_c,N_{dest})
8: **if** N_{next} is not null **then**
9: bool $i \leftarrow QO$.ENTANGLEMENTSWAPPING(N_{prev},N_{next})
10: **if** i is true **then**
11: CO.EMITCONNREQUESTTO(N_{next}, (N_c, N_{dest}))
12: **else** ▷ When the quantum operation fails
13: GRAPHUPDATE(QO, CO, N_c)
14: NAVIGATE(N_{prev}, N_{dest})
15: **end if**
16: **else** ▷ When the current node is not in the graph
17: GRAPHUPDATE(QO, CO, N_c)
18: NAVIGATE(N_{prev}, N_{dest})
19: **end if**
20: **end function**
21: CO.CONNECTIONHANDLER $\leftarrow$ NAVIGATE(N_{prev},N_{dest})

Figure 2.6. *Protocol 1. Asynchronous Routing Scheme.*

Starting from Line 4, an on-demand service is initiated to update the instant topology continuously, based on the rules specified in the distributed graph, such as Protocol 1.1 or 1.2 in the subsequent subsections. Line 5 is responsible for listening to incoming connection requests. It is worth noting that this is a self-propagating entanglement swapping process, whereby each node that receives the connection request only has to focus on connecting itself with the destination node. Once a connection request arrives, Line 21 processes it by redirecting the request to the NAVIGATE function, which begins at Line 6. The NAVIGATE function verifies if both the current and destination nodes are present in the distributed graph. If it finds that both nodes are present in the

graph, it proceeds to conduct entanglement swapping between the qubit connecting to the previous node N_{prev} and the qubit connecting to the next node N_{next}. Assuming a successful entanglement swapping, the NAVIGATE function will send a connection request to the next node, requesting the establishment of a connection between the current and destination nodes. If the entanglement swapping is unsuccessful, or either the source or the destination node is absent from the distributed graph, the node will wait for an update to the graph before attempting to navigate again.

In short, to initiate the creation of E2E entanglement between a source and a destination node, the source node issues a connection request, similar to Line 11, to begin the navigation process. Each intermediate node requests the next node along the path toward the destination to perform entanglement swapping until the destination is reached. In the subsequent two subsections, we present two illustrations of the distributed graph: DODAG and spanning tree.

2.4.1 DODAG Protocol

A DODAG [176] is a graph with a tree-like structure formed by selecting a root node and then growing by the exchange of control messages, as illustrated in Figure 2.7. The routing protocol revolves around the utilization of one or more DODAGs, employing designated *routing metrics* and an *objective function* to assist in selecting the optimal path to the root node. Subsequently, communication between the two endpoints is established via the root node. The specific graph of a DODAG at any given time is referred to as a DODAG version. If needed, a network can support multiple DODAGs with different roots.

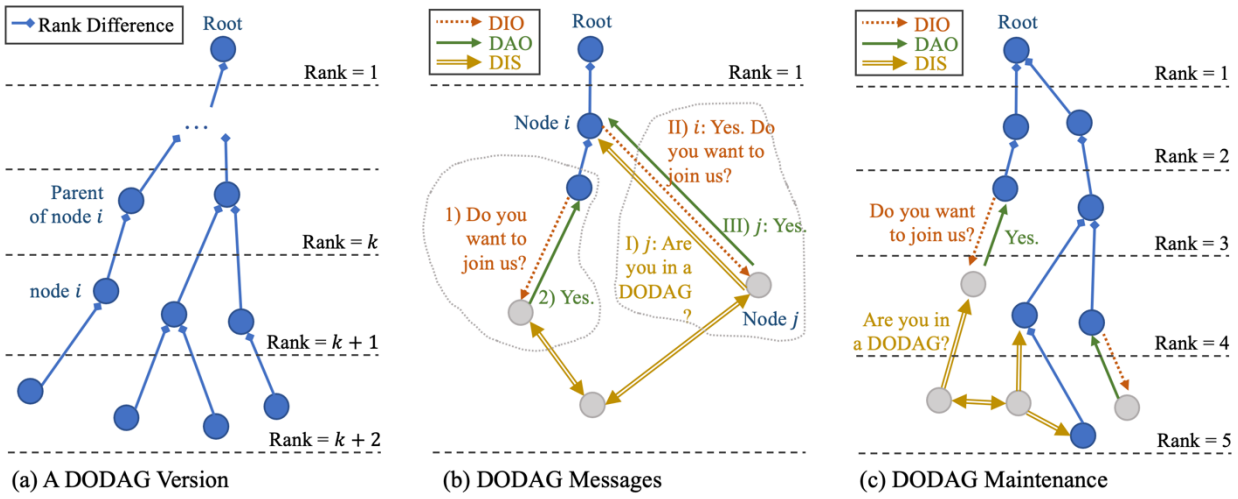


Figure 2.7. *DODAG and its Control Messages.*

Figure 2.7(a) represents a version of a DODAG at any given time. The rank values of nodes in the graph increase as their distance from the root node increases. When a node is directly connected to another node and has a lower rank value, it is considered the parent of that node. Note that the rank increment here is set as 1 for simplicity, but it can be any positive real number indicating the cost metric. In Figure 2.7(b), each node outside the DODAG communicates with its neighboring nodes by sending DIS messages to inquire if they belong to the DODAG. When a neighboring node responds with a yes (as Node i does to Node j in the figure) using DIO messages, the node (Node j) sends DAO messages asking to join the DODAG. Then, the neighboring node (Node i) computes a rank value for the new node and allows it to join. If the neighboring node does not respond, the node continues to send DIS messages until it receives a yes response or is asked to join the DODAG by a DODAG neighbor using DIO (as in the case of the dotted area on the left). In Figure 2.7(c), a DODAG expands itself and maintains its version using control messages indicated in Figure 2.7(b).

The routing metrics are the set of factors used to evaluate a candidate hop that determines a preferred path. These factors can vary, and here we include elements such as link quality, which

is determined by link capacity (C) and the probability of direct-link entanglement generation (p), as well as node quality, which is influenced by distance to the root node, number of qubits, probability of entanglement swapping (q), and coherence time (T_{co}) — these metrics aid in creating a routing strategy that maximizes the entanglement rate. To focus on the primary concern of pathfinding, we assume that node and link qualities are homogeneous. However, in actual control messages, such as DIS (DODAG Information Solicitation), DIO (DODAG Information Object), and DAO (Destination Advertisement Object) in Figure 2.7, distinct node and link qualities may be included to assist in selecting or eliminating parents and children. For example, if a link or node falls short of the minimum requirement (e.g., fidelity $f < 0.5$), it will be removed from the DODAG, along with its branches, and newer nodes will select stronger parents in their selection process. It is important to note that if a direct-link entanglement is lost (either due to decoherence or removal), the branch that was disconnected from the DODAG tree will remain a floating DODAG branch. During this time, the disconnected node will act as the temporary root node for the branch until it can successfully rejoin the main DODAG. Once it has rejoined, the floating branch will become a part of the main DODAG once again.

The objective function for a DODAG establishes how nodes select and optimize the structure of the graph. It translates routing metrics into *ranks* that determine rules such as parent selection and loop avoidance (see Appendix B). In other words, the node's rank in a DODAG indicates its connectivity with the root and is determined as a scalar measure based on the routing metrics. As the node and link qualities are considered homogenous in this chapter, the ranks should increase monotonically as the nodes get farther from the root, as shown in Figure 2.7. It is worth noting that a DODAG node can have multiple parents as long as there are no loops in the graph, which is ensured by the objective function when calculating rank for new nodes. Consequently, routing can

select preferred paths based on factors such as fewer hops or stronger nodes and links when dealing with distinct node and link qualities. An initial objective function is provided below for rank computation and parent selection.

Suppose a node v_{new} wants to join a DODAG. It must calculate the rank step between itself and each of its neighboring nodes in the DODAG, representing the amount by which the rank should be increased along a specific link. Let us represent a neighboring DODAG node as v_{nbr}, and the rank step between v_{new} and v_{nbr} as $S(v_{new}, v_{nbr})$. The value of (v_{new}, v_{nbr}) is determined by the difference between the connectivity of node v_{new} (relative to the root), denoted as $\gamma(v_{new})$, and the connectivity of node , v_{nbr}, denoted as $\gamma(v_{nbr})$. If we denote the length (i.e., the number of links) of the path from a node v to the root as $L(v)$, then $\gamma(v)$ can be calculated by:

$$\gamma(v) = p^{L(v)} q^{L(v)-1} T_{co} \qquad (2.1)$$

With that, we can compute $\gamma(v_{nbr})$ by $L(v_{nbr})$ and compute $\gamma(v_{new})$ based on the fact $L(v_{new})$ = $L(v_{nbr}) + 1$. With that, the rank step $S(v_{new}, v_{nbr})$ can be calculated as follows:

$$S(v_{new}, v_{nbr}) = |\gamma(v_{nbr}) - \gamma(v_{new})| \qquad (2.2)$$

The rank step can also be represented as $S(v_{new}, v_{nbr}) = |\gamma(v_{new})(\frac{1}{pq} - 1)T_{co}|$ since we assume uniform p, q and T_{co}. A stronger node (i.e., a node with higher connectivity) would have a smaller rank step. Next, node v_{new} selects the neighboring node with the minimum rank step (suppose it is v_{nbr}) to become its parent node. This enables node v_{new} to determine its rank value by following v_{nbr}, denoted as $\rho(v_{new})$, when it successfully joins the DODAG:

$$\rho(v_{new}) = \rho(v_{nbr}) + S(v_{new}, v_{nbr}) \qquad (2.3)$$

While a DODAG is constantly updated according to the rules above (refer to Appendix B for information on loop avoidance), the routing process in the DODAG is as simple as moving up to the root and then descending to the intended destination by selecting the parent with the lowest rank value among all parents (or both source and destination ascend to the root). The maintenance of the DODAG is outlined in Protocol 1.1 (Figure 2.8).

Algorithm 1.1 DODAG maintenance

($N_{neighbor}$: the node that the current node is communicating with)

```
1:  function NEXTHOPDETERMINATION(N_c, N_dest)
2:      return N_c.parent
3:  end function
4:  function GRAPHUPDATE(QO, CO, N_c)
5:      CO.DIS_HANDLER ← DIS(N_neighbor, CO)
6:      CO.DIO_HANDLER ← DIO(N_neighbor, QO, CO)
7:      while true do
8:          if N_c is in DODAG then
9:              CO.BROADCASTDIO(N_c)
10:             CO.LISTEN('DIS', N_neighbor)
11:         else
12:             CO.BROADCASTDIS(N_c)
13:             CO.LISTEN('DIO', N_neighbor)
14:         end if
15:     end while
16: end function
17: function DIS(N_neighbor, CO)
18:     CO.SENDDIO(N_neighbor)
19: end function
20: function DIO(N_neighbor, QO, CO)
21:     bool j ← CO.CHECKMINIRANK
22:     if j is true then
23:         CO.SENDDAO(N_neighbor)
24:         bool k ← false
25:         while !k do
26:             k ← QO.ENTANGLEMENTGENERATION(N_c, N_neighbor)
27:         end while
28:     end if
29: end function
```

Figure 2.8. *Protocol 1.1. DODAG Maintenance.*

2.4.2 Distributed Spanning Tree

As a result of having access solely to local knowledge of the instant topology of a quantum network, constructing a spanning tree — a subgraph of a connected undirected graph that covers all vertices using the least number of edges — necessitates a distributed approach. Numerous distributed algorithms for spanning tree construction have been developed, including the renowned Gallagher-Humblet-Spira (GHS) algorithm [177]. At the beginning of a GHS algorithm, each node is considered a fragment, and these fragments progressively fuse to form larger ones. To combine with another fragment, a fragment must locate its least-weight outgoing edge and work collaboratively with other nodes to find it in a distributed fashion. Over time, the number of fragments reduces until only one fragment remains, which is the Minimum Spanning Tree (MST). A spanning tree guarantees the existence of a path between any two nodes. Nevertheless, using a path will disconnect parts of the tree, requiring all nodes to update it continuously. Our assumption of uniform p, q, and T_{co} implies that all edge weights are equal in our situation.

However, we must address the instant topology's probabilistic and decoherent entanglements (links). Figure 2.9 illustrates our modified version of the GHS algorithm. In short, it addresses the separation of a fragment caused by the decoherence of entanglements and continuous attempts to generate an outgoing edge due to the probabilistic nature of entanglement generation. We achieve this by incorporating a *split* event in addition to the *merge* and *absorb* processes of the GHS algorithm. The following paragraphs outline how the modified GHS algorithm operates within a quantum network.

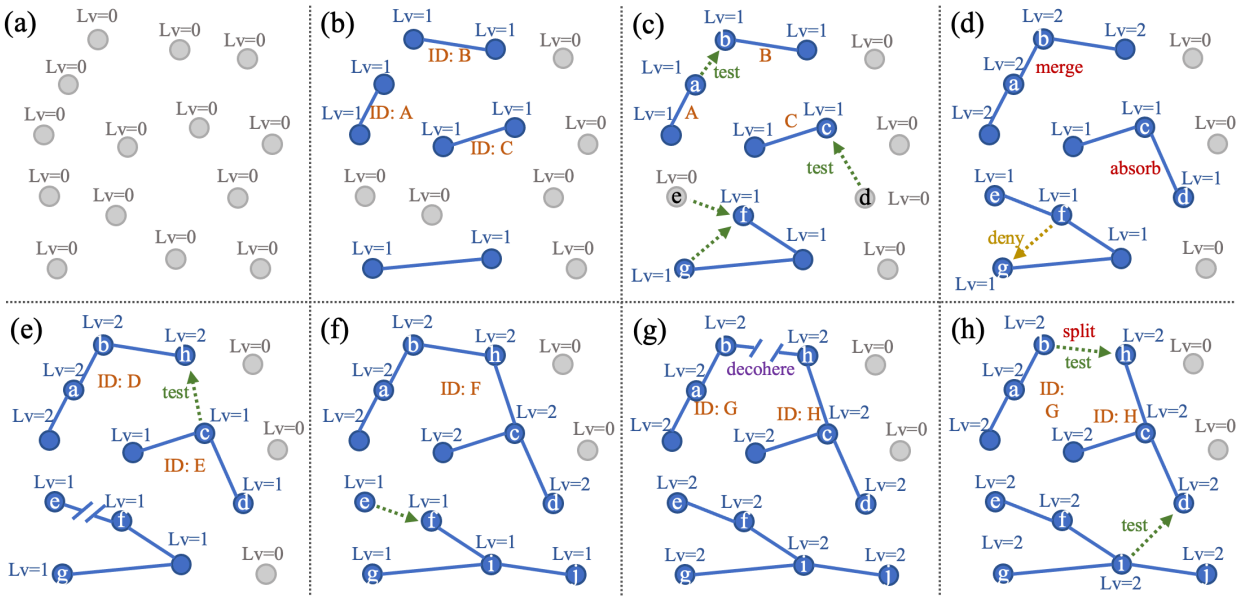


Figure 2.9. *Distributed Spanning Tree by the Modified GHS Algorithm.*

Figure 2.9 illustrates the maintenance of instant topology in a stepwise manner as a spanning tree without including the physical links. Initially, in Figure 2.9(a), all nodes are considered as fragments with a level value of 0. In Figure 2.9(b), each node sends a "test" message to its adjacent nodes, resulting in some successful and some failed connections (i.e., entanglement links). Since all level values are the same at the start, any successful connection leads to a merge process where the levels are incremented by 1, and the connecting edge becomes the core edge of that fragment. The fragments are denoted as A, B, and C. In Figure 2.9(c), nodes connected by core edges send "test" messages to their adjacent nodes (not every "test" message is shown for brevity). For example, Node a sends a message to Node b, and Node d sends a message to Node c. In Figure 2.9(d), Fragments A and B have the same level, so they merge, and their levels are incremented by 1. Fragment C absorbs another node, and its level remains the same. Note that Node f declines to connect with Node g because they have the same Fragment ID. In Figure 2.9(e), Node c sends a "test" message to Node h. In Figure 2.9(f), Fragment D absorbs Fragment E and becomes Fragment F. In Figure 2.9(g), Edge $b-h$ breaks, causing Fragment F to split into Fragments G

and H. In Figure 2.9(h), the split event is triggered, and Nodes b and h send "test" messages to their adjacent nodes.

Generically, in the physical topology $G(V, E)$, each node starts as a fragment, and the objective is to connect all nodes with minimum weight (i.e., the least number of edges in our case) with no loop. In this context, an edge represents a direct-link entanglement in a quantum network. A fragment F is a subset of V and is assigned a non-decreasing integer level $Lv(F)$ that begins with 0. Each node in a fragment must have the same level value as its fragment, i.e., $Lv(v) = Lv(F) \; \forall \; v \in F$. Fragments with a non-zero level are assigned a unique name or fragment ID, denoted as $FID(v) = FID(F) \; \forall \; v \in F$. Every node chooses a neighbor node to form a core edge under certain constraints, such as the neighbor node with the minimum weight. Each fragment has only one core edge, and the core edge selection process is beyond this text's scope but can be found in Reference [177]. The two nodes connected by a core edge are responsible for broadcasting the fragment ID and level to all other nodes in the fragment. After receiving a broadcast message from the core, every node searches for its minimum-weight outgoing incident edge. For instance, if node v receives a broadcast, it selects its minimum weight edge and sends a "test" message to node v' on the other side (assuming $v' \in F'$) with its fragment's ID and level. Then, node v' determines whether the edge is an outgoing edge and sends a message back to inform node v of the result. The decision is based on the following criteria:

1. If $FID(v) = FID(v')$, the edge is not outgoing because they are in the same fragment.
2. If $FID(v) \neq FID(v')$ and $Lv(v) \leq Lv(v')$, the edge is outgoing.
3. If $FID(v) \neq FID(v')$ and $Lv(v) > Lv(v')$, the algorithm cannot reach a conclusion because the two nodes may already belong to the same fragment, but node v' has not yet

discovered this due to the delay of a broadcast message. In this case, the algorithm allows node v' to postpone its response until its level becomes higher than or equal to the level it received from node v. Refer to Appendix C for scenarios involving broadcast delay in the GHS algorithm.

When an outgoing edge is present, the fragments will begin to merge through the merge and absorb processes (direct-link entanglement generation on the outgoing edge). If $Lv(F) = Lv(F')$, the fragments will merge through the merge process, resulting in a new fragment with an increased level of $Lv(F) + 1$. On the other hand, if $Lv(F) < Lv(F')$, the fragments will merge through the absorb process, and the combined fragment will have the same level as F'. When an edge within a fragment decoheres, it triggers a split event, where the two adjacent nodes generate new fragment IDs for their own fragments and broadcast them to the rest of the fragment. The levels of the new fragments remain the same. The increase in level during the merge process is to prevent the formation of an edge to a fragment that has already been combined by the other outgoing edge, which can cause a loop (see Appendix C). The split event does not combine fragments and thus does not affect the merge and absorb processes or the levels of any fragments.

As the rules dictate, a distributed spanning tree undergoes continuous updates. During the routing process within the spanning tree, the objective is to determine the correct direction toward the destination from the multiple available directions. Whenever an outgoing edge is detected, the relative position of each node in the tree, along with the new level and fragment ID, will be broadcast. This relative position is used to determine the correct direction toward the destination. Protocol 1.2 in Figure 2.10 provides an outline of the distributed spanning tree's update process. To be concise, we have omitted the broadcasting and listening of events that update the fragment ID and level within a fragment.

Algorithm 1.2 Distributed spanning tree

($N_{neighbor}$: the node that the current node is communicating with)
function NEXTHOPDETERMINATION(N_c, N_{dest})
    return N_c.next ▷ relative position is updated when receiving a broadcast
end function
function GRAPHUPDATE(QO, CO, N_c)
    CO.TESTMESSAGE_HANDLER ← TESTMESSAGE($N_{neighbor}$, QO, CO)
    while true **do**
        CO.BROADCASTTESTMESSAGE(N_c)
        CO.LISTEN('test', $N_{neighbor}$)
    end while
end function
function TESTMESSAGE($N_{neighbor}$, QO, CO)
    if $CO.FID(N_{neighbor}) != CO.FID(N_{neighbor})$ **then**
        if $CO.Lv(N_c) \leq CO.Lv(N_{neighbor})$ **then**
            bool $r \leftarrow$ false
            while !r **do**
                $r \leftarrow QO$.ENTANGLEMENTGENERATION(N_c, $N_{neighbor}$)
            end while
            if $CO.Lv(N_c) == CO.Lv(N_{neighbor})$ **then** ▷ merge
                char $id \leftarrow$ CO.GENERATEID
                int $lv \leftarrow CO.Lv(N_c) + 1$
                CO.SETID(N_c, id)
                CO.SETID($N_{neighbor}$, id)
                CO.SETLV(N_c, lv)
                CO.SETLV($N_{neighbor}$, lv)
            else ▷ absorb
                char $id \leftarrow CO.FID(N_{neighbor})$
                int $lv \leftarrow CO.Lv(N_{neighbor})$
                CO.SETID(N_c, id)
                CO.SETLV(N_c, lv)
            end if
            CO.BROADCASTNEWFRAGEMENTINFO(N_c, id, lv)
            CO.BROADCASTNEWFRAGEMENTINFO($N_{neighbor}$, id, lv)
        else
            TESTMESSAGE($N_{neighbor}$, QO, CO)
        end if
    end if
end function

Figure 2.10. *Protocol 1.2. Distributed Spanning Tree for Entanglement Routing.*

2.4.3 Time Complexity

The time complexity of asynchronous pathfinding algorithms can vary depending on the specifics of the algorithm and the nature of the network. Here, we analyze the time complexity of the

asynchronous protocols in terms of link availability probability p, node availability probability q, and consider a grid network with node size n (i.e., $\sqrt{n} \times \sqrt{n}$ grid) where the root is at the center of the grid.

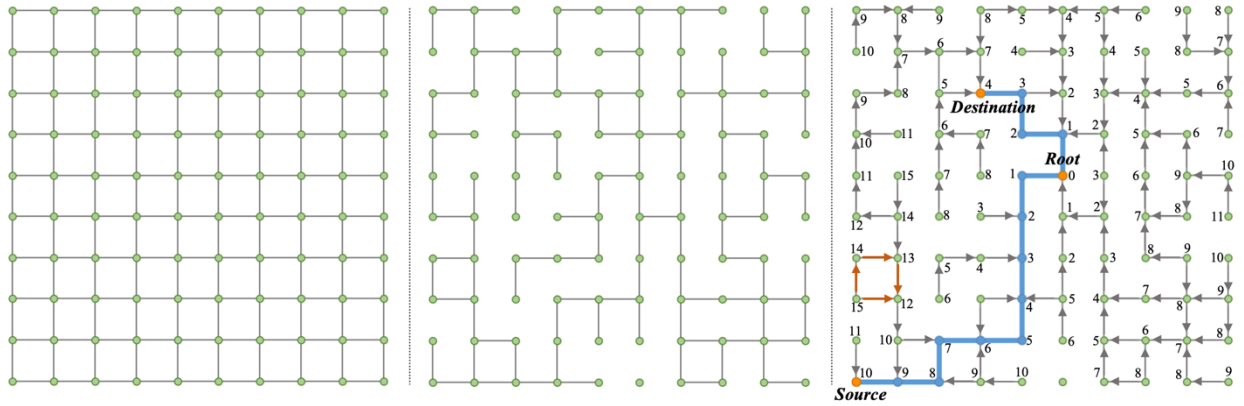
The time complexity of DODAG can be analyzed in two parts: the time complexity for constructing the DODAG and the time complexity of pathfinding within the DODAG. In the worst case, constructing a DODAG from a network can have a time complexity of $O(n)$ if we consider all possible nodes and links. However, since each link or node has a probability p or q of being available, respectively, on average, we are dealing with $p\sqrt{n}$ nodes and $q\sqrt{n}$ links (assuming a full mesh). So, the complexity becomes $O(pqn)$ for creating the DODAG. However, this decreases the probability of the pathfinding part of the protocol. In other words, the reduction of time complexity of DODAG construction of the algorithms (by the decrease of p and q) increases the success probability of pathfinding. For pathfinding, the paths from the source or destination to the root are planned by the rank values, which means the complexity corresponds to the length of the planned path. The worst-case time complexity of DODAG pathfinding is thus the summation of the furthest possible distance between source (i.e., $\sqrt{n}$) and root and that between destination and root (i.e., $\sqrt{n}$). Therefore, the overall complexity is $O(\sqrt{n})$. Note that these are the worst-case time complexities of the algorithm since we consider DODAG construction and pathfinding as separate steps, but they are happening simultaneously in practice. Thus, the true complexity can be dynamic and less than this in practice.

The time complexity of the spanning tree protocol can be similarly analyzed in two parts: constructing the distributed spanning tree and pathfinding in the spanning tree. Generally, a distributed algorithm to construct a minimum spanning tree (like the GHS algorithm) can have a

time complexity of $O(nlogn)$ [177]. In the worst case, all fragments are combined with "merge" operations, so the number of fragments decreases by half at each level. Therefore, the maximum number of levels is $O(logn)$. Each level takes $O(n)$ time (for information exchange in broadcast). Hence, the time complexity is $O(nlogn)$. Since a spanning tree is a tree, and trees are acyclic and have only one path between any two nodes, locating a path in a tree (given pointers to the start and end nodes) can be done in $O(n)$ in the worst case, as you may need to traverse the entire tree (in a skewed tree scenario). But if it is reasonably balanced, the path could be found much faster, in $O(logn)$ time. Note that the discussion of message complexity of both algorithms is out of the scope of this chapter but can be found in [178].

2.5 Performance Analysis

To assess the effectiveness of our asynchronous routing protocol, we utilize the entanglement rate (ξ) as a metric. To simplify matters, we focus on a 2D grid topology (as illustrated in Figure 2.11(a)), denoted by $G(V, E)$, and define an instant topology $G'(V', E')$ as a subset of this grid. Each node in G is only aware of the status of entanglements in the four direct links connected to it and has no knowledge of the larger instant topology beyond that.



(a) The physical topology: a grid (b) A virtual topology: a subgraph (c) A path in an instant topology

Figure 2.11. *DODAG in a Grid Topology.*

In Figure 2.11(a), a 2D grid represents the physical topology $G(V, E)$. In Figure 2.11(b), an instant topology $G'(V', E')$ is a subset of this grid. In Figure 2.11(c), a DODAG path can be identified within an instant topology, with the connected graph forming a DODAG with respect to the root node. The arrows indicate the rank differences (parent-child relationships), while small digits beside nodes indicate their rank value. It is important to note that the small squares in Figure 2.11(b) and (c) do not represent loops but occur when nodes select multiple parents, as shown in the example of orange arrows in Figure 2.11(c). These integers are solely to provide an example that illustrates the relative difference of rank values. Real values would be more fine-grained (i.e., can be any positive real numbers).

In Section 2.4.1, we used $L(v)$ to denote the length (number of links) of a path from node v to the root of a DODAG. It is possible to extend this notation to represent the length of the path between any two nodes (e.g., source node s and target node t) as $L(s, t)$, but this value only applies to a specific instance of the instant topology. To calculate the entanglement rate in a more general case, we can estimate this value by approximating the mean length of all paths between two nodes in the physical topology. Specifically, we use $l_{s,t}$ to represent the mean length of all possible paths between nodes s and t in G. $l_{s,t}$ is of interest in percolation theory for a 2D square lattice, but it cannot be determined analytically. However, an approximation is possible [179], [180] (see Appendix D).

2.5.1 Synchronous Protocol Analysis

Let us denote the average number of disjoint paths found between s and t as $n_{s,t}$, and in a 2D grid topology, we have $0 \leq n_{s,t} \leq 4$ since the maximum number of neighboring links that can be used by a node is four. Assuming we have only one possible entanglement in a direct link (i.e.,

$C/E(f) = 1$), we can estimate the average entanglement rate ξ_{syn} (per time slot) between nodes s and t for existing synchronous protocols as follows:

$$\xi_{syn}(s,t) = p^{l_{s,t}} q^{l_{s,t}-1} n_{s,t} \quad (2.4)$$

Let us denote the length of the shortest path from s to t as $\hat{l}_{s,t}$. With the range of $n_{s,t}$, we can establish the bounds for $\xi_{syn}(s,t)$ (the length of the shortest path is the minimum length for each of the four paths):

$$0 \leq \xi_{syn}(s,t) \leq 4p^{\hat{l}_{s,t}} q^{\hat{l}_{s,t}-1} \quad (2.5)$$

It is noteworthy that our examination of the entanglement rate pertains to exclusive connection requests. For an analysis of link capacity allocation in the context of a batch of requests, we refer interested readers to Reference [170].

2.5.2 Asynchronous Protocol Analysis

In essence, the idea of asynchronous protocols is to preserve the direct-link entanglements that do not fall on the route between the two communicating parties, thus enabling the upkeep of a distributed graph such as DODAG and a spanning tree for future use. Specifically, we take the DODAG protocol as an example and examine its performance in this subsection. Besides s and t, the root node of the DODAG is denoted by r. We define $l'_{s,r}$ as the average number of direct-link entanglements already in the path between the root node r and node s, and $l'_{r,t}$ as that between nodes r and t. Note that $l'_{s,r}$ and $l'_{r,t}$ are proportional to T_{co} because higher values of T_{co} result in more entanglement links that remain un-decohered in G'. With this, we can estimate the entanglement rate ξ_{DODAG} for the DODAG protocol as:

$$\xi_{DODAG}(s,t) = p^{l_{s,r}+l_{r,t}-l'_{s,r}-l'_{r,t}} q^{l_{s,r}+l_{r,t}-1} n_{s,t} \qquad (2.6)$$

To facilitate comparison, we adopt a unit time for the entanglement rate that is equivalent to one time slot in synchronous approaches. As mentioned earlier, we define the unit time as $T = \frac{T_{co}}{n}$, where $n \in \mathbb{Z}$ and $n \geq 1$. For brevity, we disregard the symbol T and express T_{co} as n unit times. Since $l'_{s,r} \propto T_{co}$ and $l'_{r,t} \propto T_{co}$, it follows that $\xi_{DODAG}(s,t)$ increases as the coherence time grows. Nonetheless, by definition, we know that $\xi_{syn}(s,t)$ is independent of T_{co}.

The range of $n_{s,t}$ in our asynchronous approach is the same as in the synchronous approach, as a node can have up to four paths to the destination (having four parent nodes). Let us denote the length of the shortest path from s to r and from r to t as $\hat{l}_{s,r}$ and $\hat{l}_{r,t}$, respectively. Then, we can establish the bounds of $\xi_{DODAG}(s,t)$ as follows (assuming s and t need to go through the root r):

$$0 \leq \xi_{DODAG}(s,t) \leq 4q^{\hat{l}_{s,r}+\hat{l}_{r,t}-1} \qquad (2.7)$$

The upper bound is attained when both s and t are already present in the DODAG, meaning no direct-link entanglement generation (expressed by p) is necessary. It should be noted that our assumption for Equation 2.7 is that s and t must pass through the root node r. However, in reality, if s and t are located on the same branch, they can pass through a shared ancestor instead. As such, the right-hand side of Equation 2.7 represents a worst-case upper bound, and in reality, the upper bounds for most specific scenarios would be greater. We demonstrate that even in the worst-case scenario where nodes need to traverse the root node, the DODAG protocol provides a higher upper bound (for most node pairs s and t) than existing synchronous approaches. On the other hand, the lower bound happens when there are no direct-link entanglements in G', and s and t cannot be incorporated into the DODAG within a single unit time.

The worst-case upper bound of our approaches is represented by $\xi^{UB}_{DODAG}(s,t) = 4q^{\hat{l}_{s,r}+\hat{l}_{r,t}-1}$. Initially, let us examine the case where $p = q$, then the upper bound of synchronous approaches is $\xi^{UB}_{syn}(s,t) = 4q^{2\hat{l}_{s,t}-1}$. Therefore, the difference between $\xi^{UB}_{DODAG}(s,t)$ and $\xi^{UB}_{syn}(s,t)$ is attributed to the difference between $2\hat{l}_{s,t}$ and $\hat{l}_{s,r} + \hat{l}_{r,t}$. Let us denote the angle formed by s, r, and t with respect to r as θ, for instance, θ_1, θ_2, and θ_3 in Figure 2.12. As demonstrated in Figure 2.12, we can infer the difference between $2\hat{l}_{s,t}$ and $\hat{l}_{s,r} + \hat{l}_{r,t}$ by the relationship of sides in a triangle:

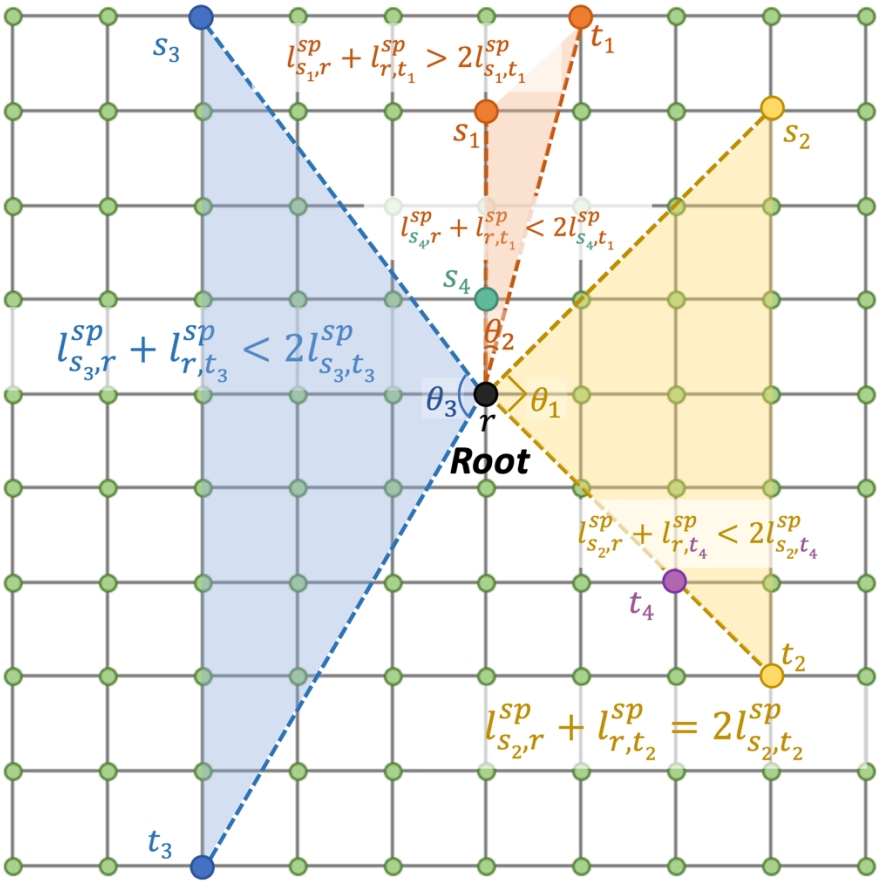


Figure 2.12. *The Location Scenarios in a DODAG.*

1. If θ is obtuse (e.g., θ_3), $\hat{l}_{s,r} + \hat{l}_{r,t} < 2\hat{l}_{s,t}$ (i.e., the sum of the two sides adjacent to an obtuse triangle is less than twice the length of the other side. See the next subsection).

2. If θ is a right angle and the three nodes form an isosceles right triangle (e.g., θ_1 with s_2 and t_2), $\hat{l}_{s,r} + \hat{l}_{r,t} = 2\hat{l}_{s,t}$. If θ is a right angle and the three nodes do not form an isosceles right triangle (e.g., θ_1 with s_2 and t_4), $\hat{l}_{s,r} + \hat{l}_{r,t} < 2\hat{l}_{s,t}$.
3. If θ is acute (e.g., θ_2), there are two cases depending on the relative position of the three nodes. Assuming s is the one closer to r (i.e., t is farther from the root. Note that these are also valid when treating t as the closer one to the root):
    a. When $\hat{l}_{s,r} \leq \alpha\hat{l}_{s,t}$ (See the next subsection for the value of α with respect to θ), $\hat{l}_{s,r} + \hat{l}_{r,t} \leq 2\hat{l}_{s,t}$ (e.g., θ_2 with s_1 and t_1). That is, our approach is better when s is close to the root but far from t. In the case of equality, the three nodes form an equilateral triangle.
    b. When $\hat{l}_{s,r} > \alpha\hat{l}_{s,t}$, $\hat{l}_{s,r} + \hat{l}_{r,t} < 2\hat{l}_{s,t}$ (e.g., θ_2 with s_4 and t_1). That is, when s is close to t but far from the root, our approach is not providing the optimal route.

In summary, only if the source and destination nodes are located in proximity and far from the root (only case 3.b above), our worst-case upper bound is lower than $\xi_{syn}^{UB}(s,t)$. Nevertheless, in this scenario, they are likely to be on the same branch of the DODAG and thus have a higher probability of having a much greater rate since they do not need to pass through the root. This hypothesis is verified through simulation in the subsequent section. Additionally, in the case where the three nodes are aligned in a line, if r is between s and t, $\hat{l}_{s,r} + \hat{l}_{r,t} = \hat{l}_{s,t}$. If r is not in between, they can always establish communication without going through the root. When traveling towards the root, the node farther away from the root would reach the closer node before reaching the root. However, any entanglement swapping between the closer node and the root would be pointless. Therefore, we can apply $\hat{l}_{s,r} + \hat{l}_{r,t} = \hat{l}_{s,t}$ in this scenario as well.

With that, it is obvious that $\xi_{DODAG}^{UB}(s,t) > \xi_{syn}^{UB}(s,t)$ holds when $p < q$.

Now, let us see the case where $p > q$. From $\xi_{syn}^{UB}(s,t)$ and $\xi_{DODAG}^{UB}(s,t)$, we have the relationship between $\hat{l}$ and $\hat{l}_{s,r} + \hat{l}_{r,t}$:

1. If $\hat{l}_{s,t} \geq \beta(\hat{l}_{s,r} + \hat{l}_{r,t})$ where $\beta < 1$ (See the next subsection for the value of β with respect to p and q), $\xi_{syn}^{UB}(s,t) \leq \xi_{DODAG}^{UB}(s,t)$.
2. If $\hat{l}_{s,t} < \beta(\hat{l}_{s,r} + \hat{l}_{r,t})$, $\xi_{syn}^{UB}(s,t) > \xi_{DODAG}^{UB}(s,t)$.

Even though the second case above seems to suggest that the synchronous method has a higher upper bound on the entanglement rate, the above relationships are valid only because $\hat{l}_{s,r} + \hat{l}_{r,t} \geq \hat{l}_{s,t}$. However, we find that, in reality, it is most likely $\hat{l}_{s,r} + \hat{l}_{r,t} \approx \hat{l}_{s,t}$ when the distance between the source and destination is large. As shown in our simulation in Figure 2.13, the paths' lengths (hops) tend to converge as the distance between the source and destination increases. That is, $l_{s,r} + l_{r,t} \approx l_{s,t}$ when the distance between source and destination is large, which means that our approach has a significantly larger upper bound than asynchronous approaches when the distance between source and destination is large. This phenomenon can be explained by Figure 2.12 to some extent (i.e., as the distance between the two communicating nodes increases, the two nodes and the root tend to align in a linear configuration if the root is in the center of the network, that is, the three nodes are more likely to form a line as distance increases. It is important to note that although the mean path length tends to decrease with larger distances between nodes, it will not be lower than the actual distance for obvious reasons. The number of network nodes influences the observed trend in a grid network. When approaching the maximum achievable distance between the source and destination, the mean path lengths for both approaches always converge. This convergence is demonstrated in Figure 2.13(a) and 2.13(b).). Also, this distance value of the

convergence point becomes smaller when T_{co} becomes larger. With this fact, Equations 2.4 and 2.6 imply $\xi^{UB}_{DODAG}(s,t) \gtrsim \xi^{UB}_{syn}(s,t)$ when the distance between source and destination is large, which makes $l_{s,r} + l_{r,t} \approx l_{s,t}$.

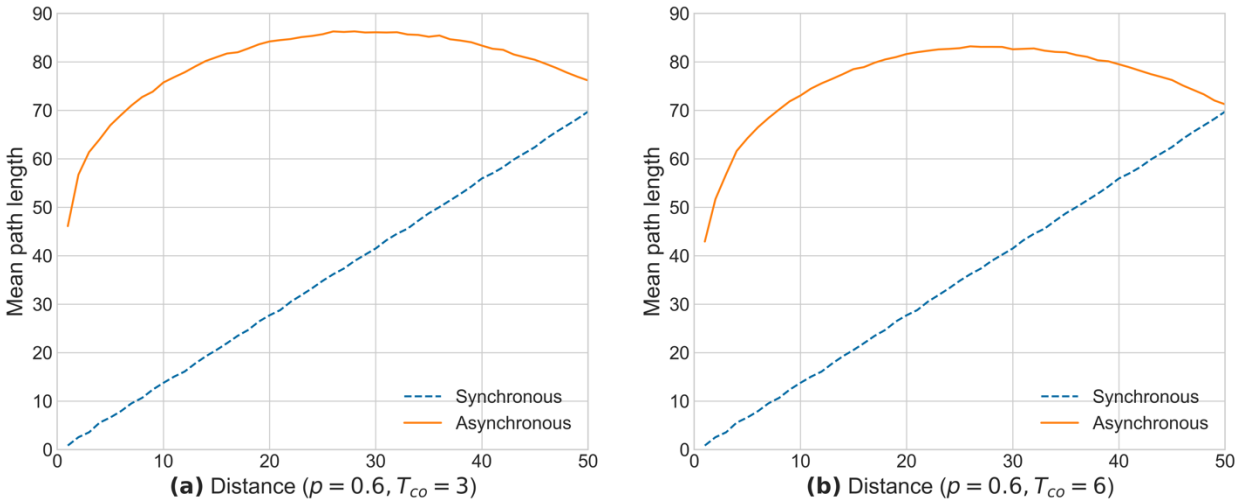


Figure 2.13. *Mean Path Length when Source and Destination must Pass through the Root.*

As shown in Figure 2.13(a), with $p = 0.6$, $T_{co} = 3$ (with source and destination locations randomly generated), the average number of hops in a path in an asynchronous approach increases rapidly when the L1 distance (see Appendix D) between source and destination is small and becomes gentle. When the distance further increases, the mean hops decrease gradually. This happens because the two nodes and the root tend to form a line (setting the root in the center of the network) as the distance between the two communicating nodes increases. It is important to note that although the mean path length tends to decrease with larger distances between nodes, it will not be lower than the actual distance for obvious reasons. This trend scales as the number of network nodes increases, and at the maximum distance between the source and destination, the mean number of path lengths for both approaches always converge. In this figure, the network size is 26×26, where the maximum distance is 50. Note that the value of q is irrelevant in this context

as the graph solely illustrates the pathfinding in an instant topology without yet reaching the phase of implementing entanglement swapping. In Figure 2.13(b), the plot with the value of T_{co} set to 6.

Note that Figure 2.13 only represents the average length of paths found by the two protocols in an instant topology. It does not consider the probability of finding a path. Path length alone does not fully determine the entanglement rate. We must also consider the probability of finding a path in an instant topology to determine the entanglement rate. In our simulations, the number of successful path findings by asynchronous protocols is significantly higher than synchronous protocols, given the same number of simulation iterations. The averages in Figure 2.13 are calculated by taking only the iterations with successful path findings. Therefore, despite potentially having longer paths, asynchronous protocols are more likely to find a path, leading to a higher entanglement rate in the end. Results pertaining to entanglement rate versus distances are presented in the next subsection.

Additionally, the simulations presented in Figure 2.13 only account for scenarios where the source and destination nodes must pass through the root to establish a connection. However, in reality (and in subsequent simulations), it is common for closely located source and destination nodes to bypass the root and instead traverse a common ancestor. As a result, the mean path length for our approach is expected to be considerably lower than depicted in the figure (as demonstrated in the simulation results in Subsection 2.6).

Moreover, if we assume that there is at least one available path (where p and q are sufficiently high to exceed the probability threshold in bond percolation, such as $p > 0.5$ and $q = 1$ for a 2D square lattice [181]), we can establish lower bounds for $\xi_{syn}^{LB}(s,t) = p^{l_{s,t}} q^{l_{s,t}-1}$ and $\xi_{DODAG}^{LB}(s,t) = p^{l_{s,r}+l_{r,t}} q^{l_{s,r}+l_{r,t}-1}$. If no direct-link entanglement is available in the instant

topology, using a DODAG approach results in a worse lower bound since it requires two paths to achieve, the sum of which is often larger than the shortest path achievable in the existing synchronous approach. However, as the distance between the source and destination increases (network scale increases), the two approaches have similar lower bounds.

2.5.3 Relative Positions of the Three Nodes in DODAG

Sides of an obtuse triangle: As shown in Figure 2.14, let us denote the sides of the triangle formed by the three nodes as a, b and c. We show that the sum of the two sides adjacent to the obtuse angle of an obtuse triangle is less than twice the length of the third side. Since it is an obtuse triangle and c is the longest side, we have

$$a^2 + b^2 < c^2 \Rightarrow (a+b)^2 < c^2 + 2ab < 4c^2 \Rightarrow a + b < 2c \qquad (2.8)$$

which means $\hat{l}_{s,r} + \hat{l}_{r,t} < 2\hat{l}_{s,t}$.

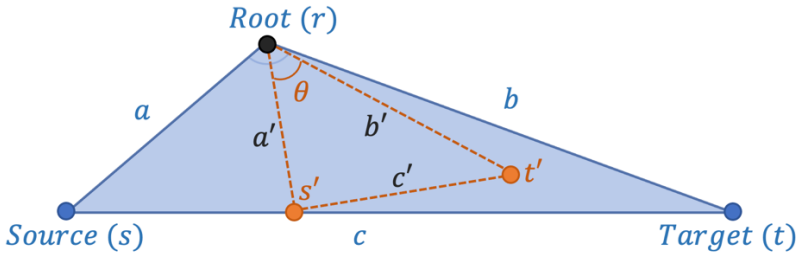


Figure 2.14. *The Triangle Formed by r, s, t.*

The value of α: As shown in Figure 2.14, we denote the sides of the triangle formed by the three nodes as a', b' and c' (when θ is acute). Based on the ASS (Angle-Side-Side) Theorem, we have

$$b' = a' \cos\theta \pm \sqrt{a'^2 - c'^2 \sin^2\theta} \qquad (2.9)$$

Together with $a' + b' = 2c'$ (for the boundary that makes the relationship of $a' + b'$ and $2c'$ different) and solve for a', we have

$$a' = c' \frac{2\cos\theta + \sqrt{2}\sqrt{2\cos^2\theta + \cos\theta - 1} + 2}{2(\cos\theta + 1)} = \alpha c' \qquad (2.10)$$

The value of β: Since we want to know the boundary between $\xi_{syn}^{UB}(s,t) = 4p^{\hat{l}_{s,t}} q^{\hat{l}_{s,t}-1}$ and $\xi_{DODAG}^{UB}(s,t) = 4q^{\hat{l}_{s,r}+\hat{l}_{r,t}-1}$ with respect to p and q. We solve for $\hat{l}_{s,t}$ by

$$4p^{\hat{l}_{s,t}} q^{\hat{l}_{s,t}-1} = 4q^{\hat{l}_{s,r}+\hat{l}_{r,t}-1} \qquad (2.11)$$

Thus, we have

$$\hat{l}_{s,t} = \left(\hat{l}_{s,r} + \hat{l}_{r,t}\right) \frac{\log q}{\log p + \log q} = \beta(\hat{l}_{s,r} + \hat{l}_{r,t}) \qquad (2.12)$$

2.6 Simulation Results

We conduct network simulations on the two approaches to support our observation of the theoretical performance analysis. We conduct simulation with single-path scenarios using NextworkX [182], where we only find one optimal path for the source and destination in each unit time. We did the single-path scenarios since it is much faster to simulate in a large grid topology. A multi-path scenario is also shown at the end. The locations of the source and destination are generated randomly. The root node for DODAG is set at the center of the grid. To simulate the probabilistic behaviors of direct-link entanglement generation, we employ a grid network with edges that have assigned probabilities for availability. Subsequently, we simulate the entanglement swapping process within the probabilistic vertices of the network. Within this simulated grid network, we conduct 50,000 iterations of each protocol for random pairs located at the same distance, aiming to achieve E2E entanglement.

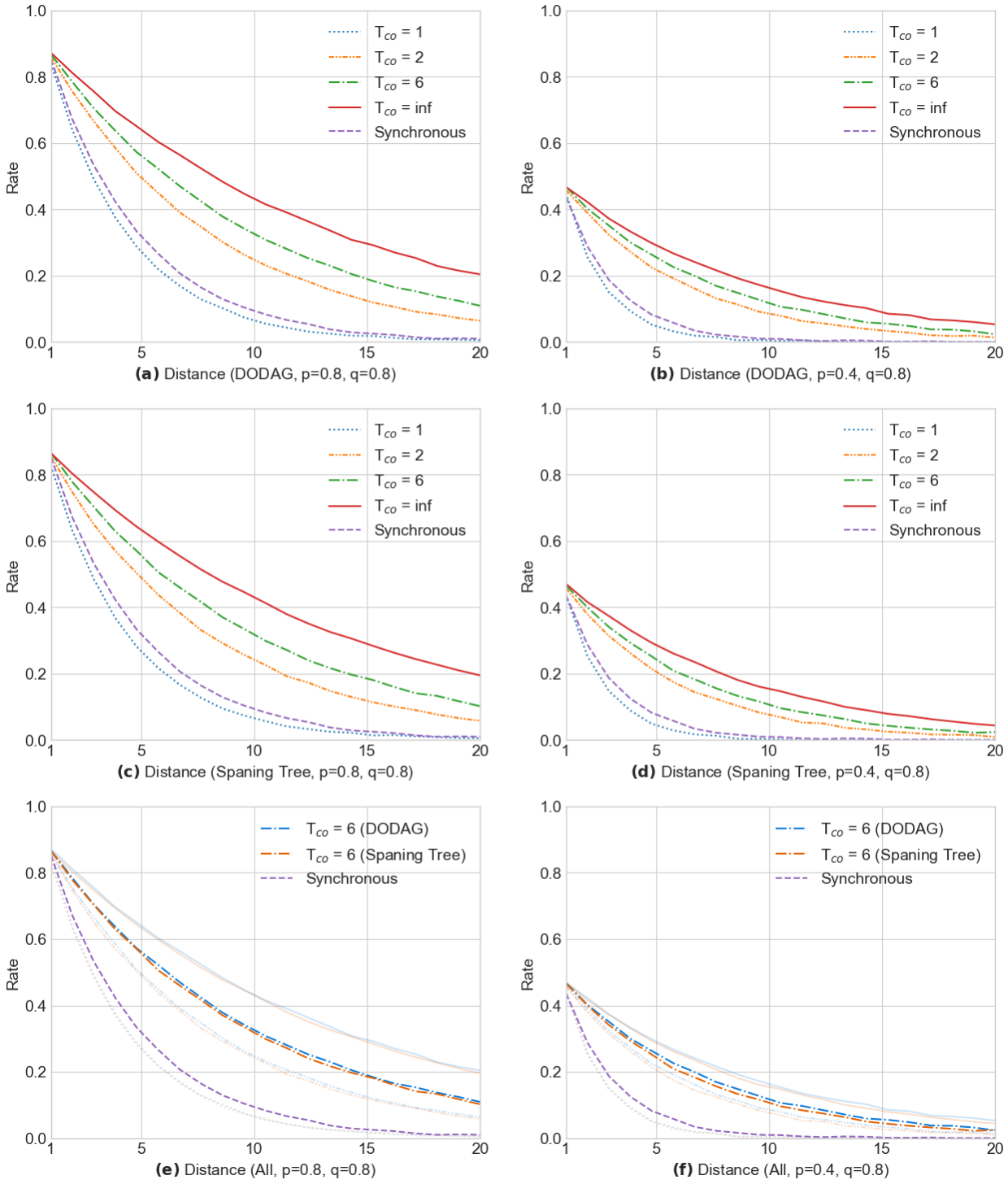


Figure 2.15. *Simulation Results: Rate vs. Distance.*

Figures 2.15(a), (c), and (e) show the entanglement rate behaviors when $p = q = 0.8$, and Figures 2.15(b), (d), and (f) show those when $p = 0.4$ and $q = 0.8$ (with local knowledge of the instant

topology). The first row, represented by (a) and (b), showcases the DODAG performance, while the second row, (c) and (d), highlights the spanning tree performance. These are compared to the performance of synchronous protocols, denoted by the dashed purple lines. The third row, (e) and (f), includes all protocols together and highlights asynchronous ones when $T_{co} = 6$.

As shown in Figure 2.15, the entanglement rates of asynchronous protocols (both DODAG and spanning tree) decay slower than synchronous approaches. We can tell that DODAG and spanning tree approaches behave similarly (DODAG slightly better) in simulation. Both provide higher rates than synchronous approaches when the coherence time is greater than one unit time. We try different combinations of p and q in the simulation. In all combinations, asynchronous protocols have a higher entanglement rate and decay slower than synchronous ones when the coherence time is greater than one unit time. When coherence time is equal to one, asynchronous protocols perform similarly to synchronous protocols when the distance is small but slightly worse when the distance increases. The solid lines in the figure show the estimated upper bounds of the entanglement rates of an asynchronous protocol under the particular parameter settings, which is obtained by using infinite coherence time. The figure also shows that asynchronous protocol has a higher rate even in short-distance scenarios. This verifies our speculation in Subsection 2.5.2 that close-by node pairs have a high chance of not needing to go through the root but only through a common ancestor. Note that the simulation results represent the average rate across multiple iterations, which introduces some inherent noise. Additionally, intermediate entanglements (i.e., the intermediate entanglements established towards E2E entanglements) set up through entanglement swapping will refresh the coherence time and are retained in asynchronous routing processes. Consequently, the rates in simulations at distance d may exceed q^d when using asynchronous protocols.

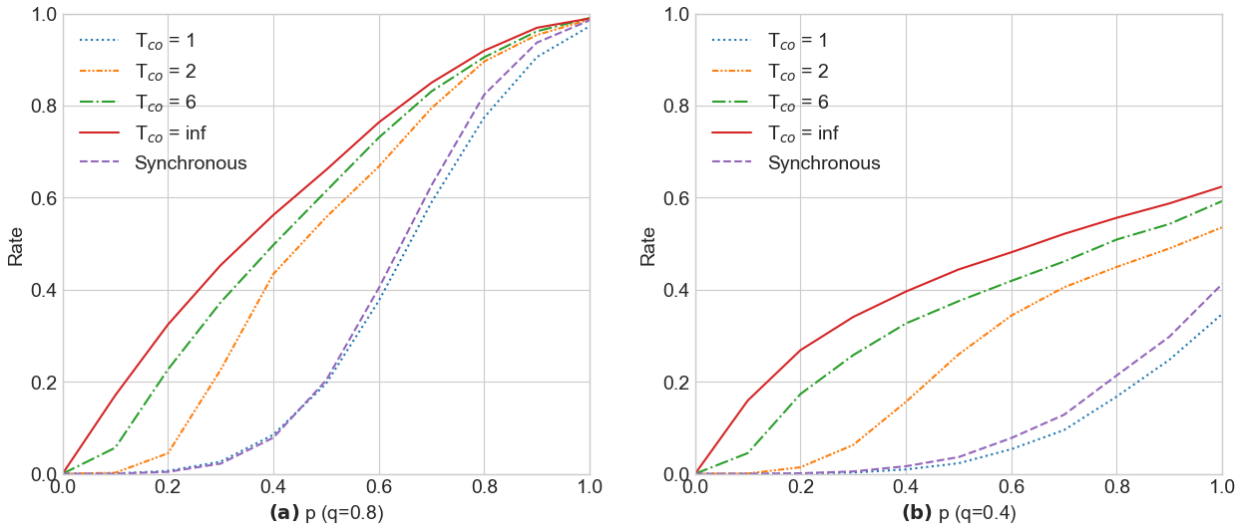


Figure 2.16. *Simulation Results: Rate vs. p.*

The two diagrams in Figure 2.16 show the relationship between entanglement rate and p in the two types of protocols. For $q = 0.8$, this relationship is depicted in Figure 2.16(a), and for $q = 0.4$, it is presented in 2.16(b). The coherence time T_{co} is varied across the set $\{1, 2, 6, \infty\}$. When the coherence time is greater than one, the asynchronous protocol has higher rates in all values of p. When the coherence time equals one unit time, the asynchronous protocol performs similarly to the synchronous protocol, and its entanglement rate is slightly smaller as p increases. In both situations, the entanglement rate of asynchronous routing increases as coherence time increases, but the synchronous protocol does not.

As shown in Figure 2.16, the entanglement rate of asynchronous routing increases as coherence time increases. The performance of existing synchronous protocols such as [116] is independent of the coherence time. This indicates that when technology improves, asynchronous protocols will have a much more impact than synchronous protocols. Since the performance of DODAG and spanning tree protocols is similar in simulation, we only show the DODAG graph in this figure for concise. The plot also demonstrates the estimated upper bound of the asynchronous protocol under

a particular q when the coherence time is infinite. For example, the upper bounds of the entanglement rate of an asynchronous protocol when $q = 0.8$ or $q = 0.4$ are shown with the solid line in Figure 2.16. Nonetheless, if we have $p = q = 1$, we do not need any asynchronous or synchronous protocols running in the case of having local knowledge of the instant topology. With that, we can run any shortest path algorithm using the physical topology. However, asynchronous protocols will still be useful, considering a quantum mobile network with dynamic physical topology (i.e., nodes may be moving). Moreover, with $T_{co} = 1$, the curves of asynchronous and synchronous protocols look just like the percolation probability in percolation theory and get increasingly obvious (more and more rapid transition around the threshold) when q gets larger.

In this chapter, we equate quantum state lifetime with the coherence time of entanglement, considering them synonymous due to the dependency of entanglement coherence time on qubit lifetime. As entanglement is sustained by quantum states, its coherence period is determined by how long the states remain coherent. Additionally, we posit that the usable lifetime of qubit—its duration of reusability for entanglement generation—is sufficiently long. Utilizing unit time, which conveys a relative rather than absolute value, prevents us from specifying a minimum quantum state lifetime, prompting us to assume it to be suitably long.

Finally, we show the simulation results in multi-path scenarios (Figure 2.17). Multi-path simulation means we search for more than one path (disjoint) for a pair of source and destination. We can tell asynchronous protocols have a higher entanglement rate than synchronous protocols in all scenarios when $T_{co} > 1$.

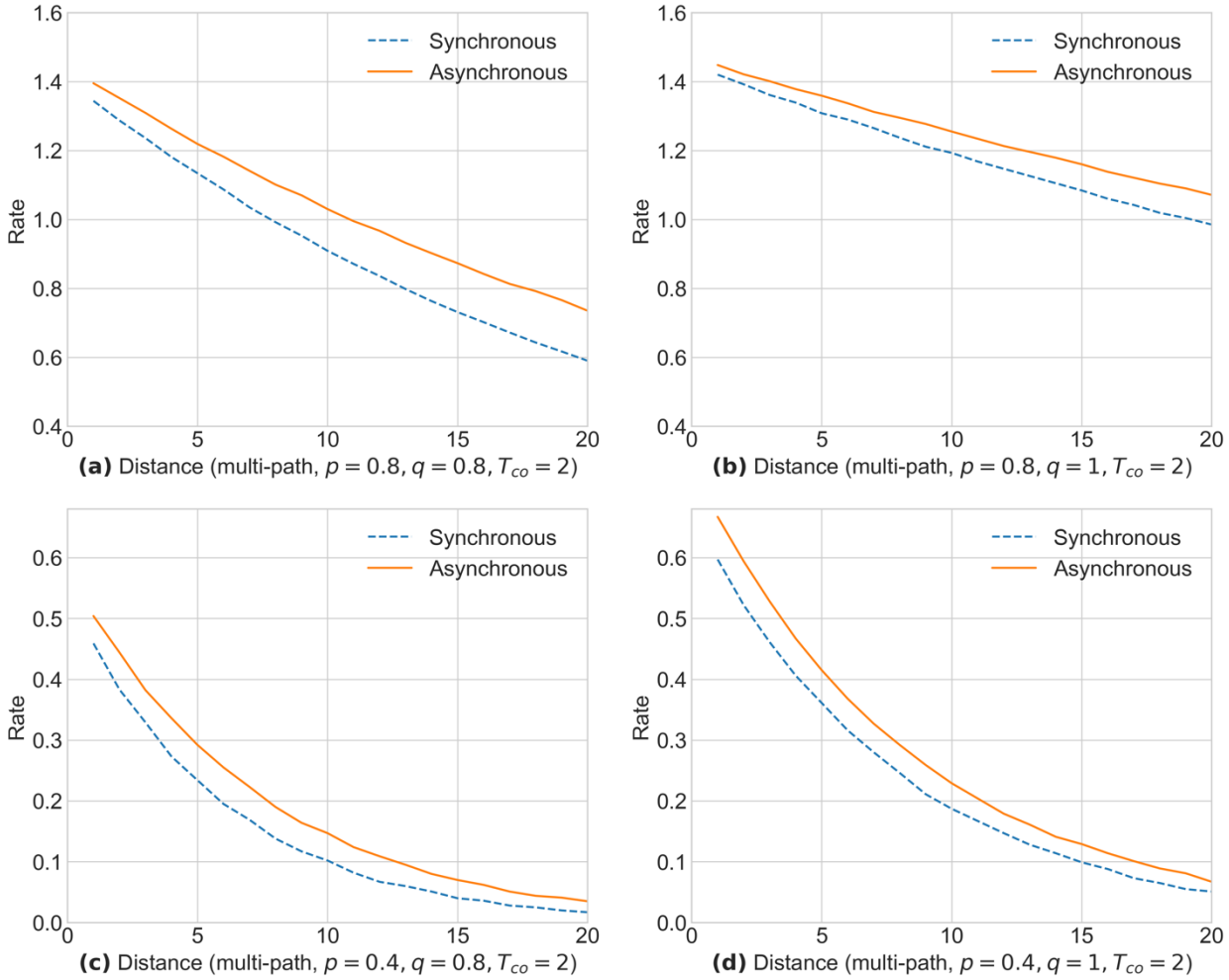


Figure 2.17. *Multi-path Simulation Results: Rate vs. p.*

Previous diagrams demonstrate the performance of single-path scenarios. Figure 2.17 shows the performance comparisons of multi-path scenarios of the protocols. In varying scenarios with different p, q, and T_{co} values, asynchronous protocols consistently outperform synchronous ones in terms of entanglement rate. Specifically, for $T_{co} = 2$ and $p = q = 0.8$, the result is shown in Figure 2.17(a). The scenarios "$p = 0.8, q = 1$," "$p = 0.4, q = 0.8$," and "$p = 0.4, q = 1$" are illustrated in 2.17(b), (c), and (d), respectively. Overall, our protocol provides larger rates than synchronous approaches in all scenarios when $T_{co} > 1$. Note that when $q = 1$ and $q > 0.5$ (the bond percolation threshold for the square lattice [180]), the distance affects less on the rate. The rate is not distance independent (it actually is when considering only one path) because there are

fewer alternative paths as distance increases. This behavior conforms to what is expected in percolation theory.

With the simulation results, we can establish the contributions of our asynchronous protocols:

1. The entanglement rate of asynchronous routing increases with the coherence time.
2. The entanglement rate of asynchronous routing decays slower than synchronous approaches.
3. The entanglement rate of asynchronous routing is larger than synchronous approaches in almost all scenarios.

2.7 Summary

We envision quantum-native routing (i.e., routing for E2E entanglement) as the key component of the quantum internet and identify the disadvantages of synchronization used in existing quantum-native routing schemes. Thus, we proposed the design of the asynchronous routing scheme and benchmarked its performance under different parameter settings. Here, we highlight the differences between our asynchronous routing protocols and existing asynchronous protocols. Under the condition that a node only knows the direct-link entanglement states of its adjacent links, existing approaches use synchronized time slots, including two execution phases to collaborate all nodes to search for a path connecting the source and destination nodes. This method consumes all direct-link entanglements, trying to find multiple paths between them. However, the consumption of all direct-link entanglements counters the entanglement rate. Our method removes such synchronization of execution phases by introducing graph maintenance in a distributed manner. We treat all network nodes as vertices in a graph that try to connect their neighbors using direct-link entanglements. No central control and no global knowledge of the direct-link entanglements

are needed in our method. Whenever a pair of nodes wants to create an E2E entanglement, they go through the distributed graph that is continuously updated to find each other. With analysis and simulation, we found that the entanglement rate of our asynchronous protocols increases with the coherence time, decays slower than and is generally larger than that of asynchronous approaches. Furthermore, asynchronous protocols can simultaneously handle multiple user pairs, which remains an open problem in synchronous schemes. Exploring the performance of multi-user scenarios holds significant prospects for future research.

Moreover, existing approaches treat repeater nodes equally. However, repeaters have different performances (they can have weights for edges, e.g., transmissivity, for routing, but no preference for using competent nodes for more important purposes). Even though this is not the primary purpose, DODAG can reflect the node robustness by preferring to set robust nodes as root nodes. Also, DODAG roots can serve as well-fit gateways to connect multiple networks (i.e., networks of networks). Since DODAG has such benefits that a spanning tree does not have and has similar performance, we recommend DODAG to be the primary protocol for asynchronous routing, which is also well-studied and standardized for low-cost wireless networks. Additionally, the scalability of the DODAG network can be enhanced by constructing hierarchical DODAGs, which involve creating trees of trees. This direction holds potential for future exploration.

There are other aspects of the protocol that warrant further investigation. For example, the center of a network is a good position for the root node, but the impact of different root positions remains future work. Moreover, the strategies to arrange multiple roots' positions and to determine how fine grain (the density) the partitioned networks should be (distance between adjacent root nodes) remain future work. Last but not least, other aspects of entanglement-swapping-based routing are also important and under investigation, such as the use of untrusted repeaters, the minimal trust

between networks (i.e., no knowledge of the internals of another network), the possible presence of malicious nodes, and the order of entanglement swapping along a path, etc. Furthermore, exploring the performance of asynchronous protocols in non-grid networks is worthwhile. Through this work, we aim to raise awareness and foster advancements in networking techniques for the quantum internet.

Chapter 3: Multi-tree Quantum Routing

In Chapter 2, we present AER, which enhance entanglement rates by leveraging solely the local knowledge of the node's entanglement links. It eliminates synchronous operations and conserving unused entanglement links. In this chapter, we present a multi-tree approach with multiple DODAGs designed to improve E2E entanglement rates in large-scale networks, specifically catering to a range of network topologies, including grids and barbells, as well as realistic topologies found in research testbeds like ESnet and Internet2. Simulations show a marked improvement in E2E entanglement rates for specific topologies compared to the single-tree method. This study underscores the promise of asynchronous routing schemes in quantum networks, highlighting the effectiveness of asynchronous routing across different network topologies and proposing a superior routing tactic.

3.1 Introduction

Advancements in quantum technology have ushered in new horizons for communication networks [30], [1]. Quantum networks, underpinned by the principle of quantum entanglement, present the potential for unparalleled security and computational advantages over classical counterparts. Pioneering applications, such as QKD and DQC, are a testament to these capabilities. Essential to these applications is creating E2E entanglement over long distances, also known as entanglement distribution. This challenge is being addressed by the development of quantum repeaters. These repeaters employ entanglement swapping, enabling two quantum nodes that are not directly connected to establish entanglement via an intermediate node, i.e., the repeater. This procedure can be extended over multiple repeaters, creating a long-distance entanglement link.

As mentioned in Chapter 2, in prevalent methods such as [115], [116], the routing procedures bifurcate into two distinct phases: the external and the internal, each with a designated time slot. In the external phase, node pairs with entangled photons create a direct link between them. This results in an instant topology of these successfully established direct-link entanglements. Figure 2.2(a) shows an initial state (physical topology) which results in several direct links, e.g., E_{AD} shown in Figure 2.2(b). Transitioning to the internal phase, under the presumption that nodes are only aware of the generation outcomes of the direct-link entanglements with their adjacent nodes (referred to as local knowledge of the instant topology), each repeater node indiscriminately swaps entanglements, aiming to establish a multi-hop entanglement between two communication-requesting endpoints as illustrated in Figure 2.2(c) and 2.2(d). E_{AC} in Figure 2.2(d) is an example. If the nodes A and C were wanting to communicate, this would be the desired E2E entanglement. Should this E2E entanglement fall short, the two phases would repeat. To maintain the availability of entanglement links for the internal phase, synchronizing both phases becomes imperative. Yet, such synchronization and indiscriminate trials expend all available entanglements within each timeframe, resulting in a diminished E2E entanglement rate.

In Chapter 2, we proposed an asynchronous routing protocol that identifies paths between any two endpoints at a substantially enhanced entanglement rate compared to the synchronized two-phase methods. Additionally, as quantum technology advances, our rate is expected to increase along with extended coherence times. Moreover, the incorporation of root nodes as gateways can facilitate the integration of multiple interconnected networks. By iteratively employing an instant topology built on entanglement links, our approach obviates the need for synchronized operations. This instant topology can manifest as a tree-like graph akin to a DODAG or a spanning tree. Our theoretical evaluations and simulations showed that asynchronous methods outpace their

synchronous counterparts by achieving a larger upper bound and an increased E2E entanglement rate.

While the asynchronous routing scheme has introduced a practical approach using a single tree-like structure, relying on a single DODAG may introduce limitations, especially regarding scalability and resilience in an expanding quantum network landscape. As networks grow in complexity and size, there arises a need for multiple interconnected trees to achieve better routing efficiency, fault tolerance, and quicker E2E entanglement generation. In this chapter, we propose the integration of multiple trees (multiple DODAGs) for quantum network routing and highlight its advantages over the single-tree approach and synchronous methods. Our simulations indicate a significant enhancement in E2E entanglement rates for specific topologies, including grid, barbell, and some realistic testbed structures using the multi-tree approach. We also identify the optimal graph types for tree routing schemes, which feature evenly distributed nodes with more uniform and higher connectivity. This research emphasizes the potential of asynchronous routing methods in quantum networks, spotlighting the success of asynchronous routing in various network structures and advocating advanced routing strategies.

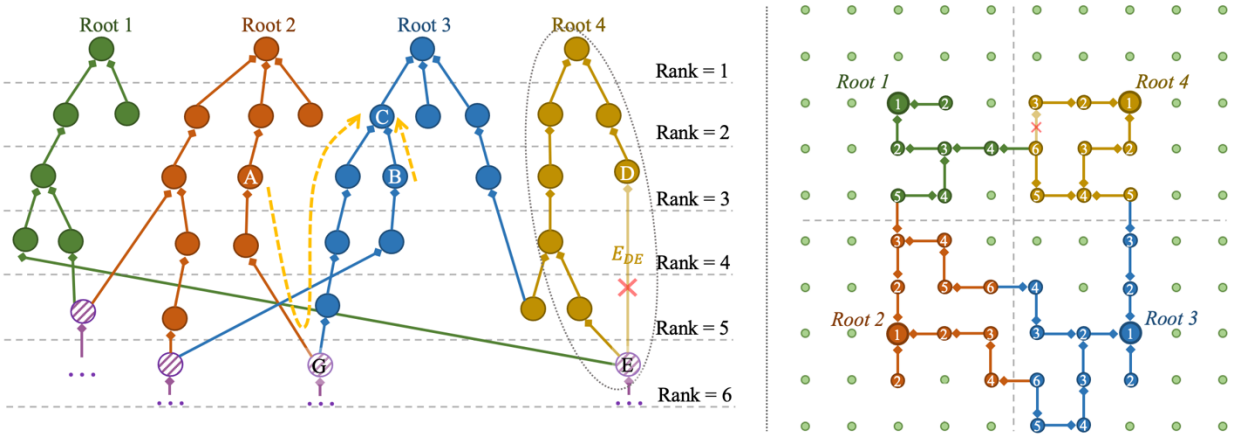
3.2 Multi-Tree Entanglement Routing

With the same background and motivation of Chapter 2, this section explores integrating a multi-tree structure with DODAG in our asynchronous routing framework, addressing the root node selection and the multi-tree applicability.

3.2.1 Multi-tree Structure

A multi-tree structure comprises several trees that may share subtrees while avoiding the creation of loops. When combined with DODAG, it forms a collection of DODAGs with nodes organized

hierarchically based on their rank values, as illustrated in Figure 3.1. Each DODAG generates its own tree, originating from its root node. When two leaf nodes intersect, they engage in a negotiation to determine which one will become the parent, typically by comparing their rank values. The leaf node with the smaller rank value is selected as the parent. Nodes with identical rank values are called "comparable nodes" and do not establish connections with each other. The subtrees that emerge after the patterned purple nodes in Figure 3.1(a) represent the shared subtrees.



(a) A multi-tree structure by DODAGs (b) A grid topology forming the structure in (a)

Figure 3.1. *A Multi-tree Structure: a DODAG Forest Instance and its Representation in a Grid Topology.*

The hierarchical node arrangement by rank value eliminates the possibility of loop formation inside one tree. Also, diamond patterns are prohibited to avoid loops between trees. For example, the subgraph outlined by the dotted ellipse in Figure 3.1(a) demonstrates this prohibition. A diamond pattern occurs when a node with parents from multiple trees has more than one parent within a single tree. This cannot be allowed because when a source node seeks a destination path, it explores both lower and higher-ranked directions, creating an undirected crosstree path. This undirected path would lead to loop formation during the pathfinding phase [183]. To prevent these patterns, nodes refrain from being overly selective when choosing parents. In other words, a node should only select one parent per tree. For example, in Figure 3.1(a), if Node E already has a parent

in the fourth tree, it should abstain from creating another entanglement in the tree, such as E_{DE}, with Node D.

In this context, finding a path between nodes entails identifying a shared parent, with a preference for selecting lower-ranked parents. The search extends to higher-ranked nodes if no common parent is located among the lower-ranked nodes. For instance, consider Node A in Figure 3.1(a), aiming to establish entanglement with Node B. The path for E2E entanglement is indicated by the yellow dashed arrows. Initially, it attempts to locate Node B among its lower-ranked peers but fails. Then, Node A extends the search to higher-ranked nodes, seeking a child node with an outgoing link to another tree. Node A finally identifies Node G and routes through it to reach Node C, which serves as a common parent for both Nodes A and B. This process is indicated by the dotted yellow arrows, signifying the undirected path from Node A to Node C and from Node B to Node C. It is worth noting that this pathfinding operates using classical methods. Once the path is established, entanglement swapping along the path is executed to create the E2E entanglement.

It is worth noting that the path search stops upon reaching the destination in most cases, but in the worst-case scenario, the source node may explore the entire forest. Even then, it only provides multiple paths for selection, not attempting swapping throughout. The source can choose the shortest one and perform swapping accordingly. The exponential decay of performance occurs during swapping, not during classical communication. However, in the worst-case scenario, most methods, including ours, may encounter issues with classical communication overhead in large networks, where no perfect solution exists. In the Noisy Intermediate-Scale Quantum (NISQ) era, we focus on optimizing performance despite these noisy conditions.

The first topology usually considered in routing is that of a grid. In a typical grid topology, nodes are neatly surrounded by neighboring nodes, typically four: north, south, east, and west. With that, the initial step involves the selection of root nodes, ideally one per quadrant in larger grids, while a central root may suffice for smaller grids. These roots serve as primary sources of information flow. Each layer of trees extends outward from these roots, including nodes at increasing distances. Due to the grid's symmetry, this arrangement results in concentric trees of nodes around each root. Figure 3.1(b) shows an example of four trees in a grid topology corresponding to the one depicted in Figure 3.1(a).

The inherent predictability of grid layouts and the structured deployment of DODAGs minimizes the occurrence of lengthy, convoluted routes, especially for nodes situated far from one root but closer to another root within a multi-tree structure. Consequently, shorter routes reduce probabilistic operations, such as entanglement swapping, which, in turn, contributes to an increased entanglement rate.

However, employing multiple DODAGs within a grid also presents challenges. The overlapping zones of neighboring DODAGs can lead to communication overheads as nodes actively negotiate parent selection. This issue can be mitigated by discouraging the greedy selection of parents in different DODAGs or using priority and scheduling mechanisms.

3.2.2 Root Nodes Selection

The root node selection is crucial in optimizing entanglement distribution and minimizing overhead in any network topology, including multiple or single-tree schemes. For example, central nodes often have equidistant access to all other nodes in the grid, ensuring an even distribution of

resources and minimizing the maximum path length to any node. Corner nodes allow for a clear directional flow of information across the grid.

Regarding a multi-tree structure, selecting roots in denser areas of the network is preferable. These areas, characterized by higher interaction rates, allow for more efficient entanglement swapping due to the closer proximity of nodes. Clustering the network by node density and designating the centers of these clusters as root nodes can significantly improve the network's entanglement distribution. These central cluster nodes serve as effective hubs, managing local traffic efficiently and reducing the need for failure-prone or long-distance connections.

With that, considering the eccentricity of nodes is another pivotal factor in the root node selection process. Eccentricity is the maximum distance between a node and any other node in the network. Nodes with low eccentricity are preferable as root nodes because they are more centrally located, thereby reducing the maximum number of entanglement swapping must occur. Moreover, nodes with high connectivity are excellent candidates for root status. Their multiple adjacent edges imply an ability to handle numerous entanglement requests and to support a high volume of entanglement-swapping processes. A high-connectivity node as a root also adds robustness to the network, offering multiple alternative pathways for entanglement distribution in case of individual link failures.

Dynamic root selection strategies can also benefit the network by adapting to evolving conditions such as traffic patterns or connectivity changes. Nodes that become central to the network's traffic flow areas can be dynamically chosen as root nodes to optimize distribution routes (if the instant topology allows). Additionally, periodic rotation of root nodes can mitigate the risks of network bottlenecks and single points of failure. It also distributes operational stress evenly across the

network, which is especially important if the role of a root node causes any form of degradation or wear over time.

3.2.2 Applicability

The use of multi-tree structures offers advantages in routing entanglement. However, the specific network topology influences the choice to employ multiple trees. While multi-tree structures can be applied to various network topologies, some network configurations benefit more from this approach.

Multi-tree routing is generally more effective in larger, more regularly distributed, or densely connected topologies than linear chains or backbones. Its goal is to facilitate nodes in proximity within a large network without routing through a central root. It enhances entanglement rates for nearby nodes while maintaining rates for farther nodes. These findings are supported by the simulations presented in the next section.

Grid topologies, characterized by even connectivity, are well-suited for multi-tree routing. Using multiple trees in such grids ensures balanced load distribution. In larger grid networks, adopting a multi-tree structure significantly reduces the path length needed for E2E entanglement.

Conversely, multi-tree routing may not be the most competitive in barbell topologies, but it still outperforms a single-tree approach. In this scenario, root nodes can be placed at the center of each side of the barbell. However, the backbone link connecting the clusters needs more resources and management than links in more evenly distributed topologies. Challenges also arise in topologies with long linear chains, posing difficulties for entanglement routing for all schemes.

3.3 Simulation Results

This section presents our simulation settings and results.

3.3.1 Simulation Settings

We have developed simulation models for multi-tree, single-tree, and synchronous routing schemes across three generated topologies: grid, random graph, and barbell. In all experiments, we maintain a constant direct-link entanglement generation success rate (p) and entanglement swapping success rate (q) at 0.8, along with a coherence time (T_{co}) of 2, which facilitates the comparability of results in various scenarios. Requests are started by choosing a source node at random and then selecting a destination node at random (across the entire network) from the group of nodes at the specified graph distance away.

The grid graph comprises a ten-by-ten network. In the case of multi-tree routing, we establish four root nodes positioned at the center of each quadrant, mirroring the configuration presented in Section 3.2. Conversely, a single root node is designated at the center of the entire grid for single-tree routing.

A random topology graph is generated using the Erdős-Rényi model, also known as a binomial graph [184]. This model assigns probabilities for edge creation within the graph. Note that these probabilities only relate to the generation of physical topology. To ensure a node degree like the grid for comparison, we generate a random network with 100 nodes using a probability of 0.04. This yields an average of four adjacent edges per node.

The barbell topology comprises two clusters linked by a single connection. Each cluster constitutes a random graph containing 50 nodes, with an average of four edges per node. In the case of multi-

tree routing, both clusters are further divided into two subgraphs, resulting in four root nodes. A single center is located at one end of the backbone link for single-tree routing.

The compared synchronous scheme relies on the two phases outlined in Section 3.1 and [116]. It operates on local knowledge of the instant topology. During the internal phase, each node assists in creating E2E entanglement for a user pair. Nodes keep a distance table and perform swapping operations for nodes closer to the source or destination. This synchronized approach involves exhaustive trials, utilizing all available entanglements.

3.3.2 Results for Generated Topologies

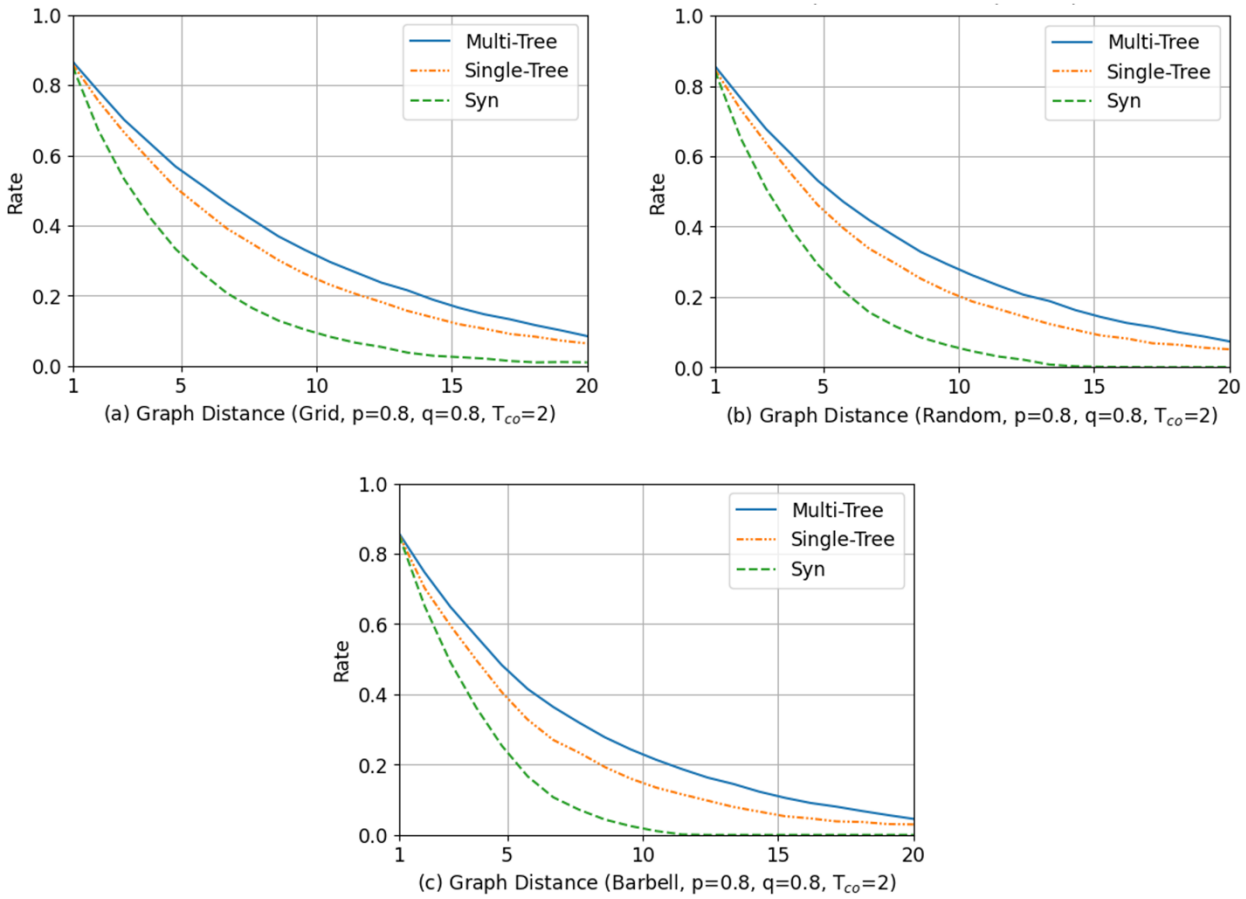


Figure 3.2. *Simulation Results for Generated Topologies.*

Figure 3.2 shows the entanglement rates for three routing schemes—multi-tree, single-tree structures, and synchronous (i.e., Syn) across three different network topologies—grid, random graph, and barbell over various distances. Graph distance is measured by the shortest path length between two nodes in the corresponding physical topology (i.e., hops). This chapter focuses on the impact of network topology rather than the actual distance and thus does not simulate p as a function of distance. The mean entanglement rate is computed by dividing the number of successful E2E entanglements by the number of attempts.

The multi-tree approach shows advantages in the grid topology, characterized by its regular and evenly distributed layout. Conversely, a single-tree structure within a grid topology may lead to longer paths for nodes far from the central root node. In the random graph topology, nodes exhibit connectivity similar to the grid topology but are less evenly distributed. Therefore, they behave similarly to the grid topology but perform slightly worse. In the barbell topology, nodes within the same cluster enjoy better connectivity, but performance relies on the backbone link between clusters. The entanglement rate is substantially lower for pairs of nodes located on different sides of the barbell.

The “Syn” curve in Figure 3.2, representing a synchronous routing scheme, typically exhibits lower rates than asynchronous multi-tree and single-tree schemes. This is because synchronous schemes may rely on coordination and blind searches, which may not scale well with distance, especially in less regular topologies. Overall, the improved performance of the multi-tree approach over the single-tree structure is attributable to node pairs in the same cluster, i.e., closer nodes.

3.3.2 Simulations with Realistic Topologies

To validate the effectiveness of the routing schemes in real-world scenarios, we also conducted simulations of the three routing methods on three actual research networks across the globe: ESnet [185], SURFnet [186], and Internet 2 [187]. These networks were explicitly selected to showcase the variance in graph structures, from a more uniformly structured SURFnet to a more linear configuration of Internet2. Note that distance in this context still refers to the shortest path length between two nodes in the physical topology rather than geographical distance.

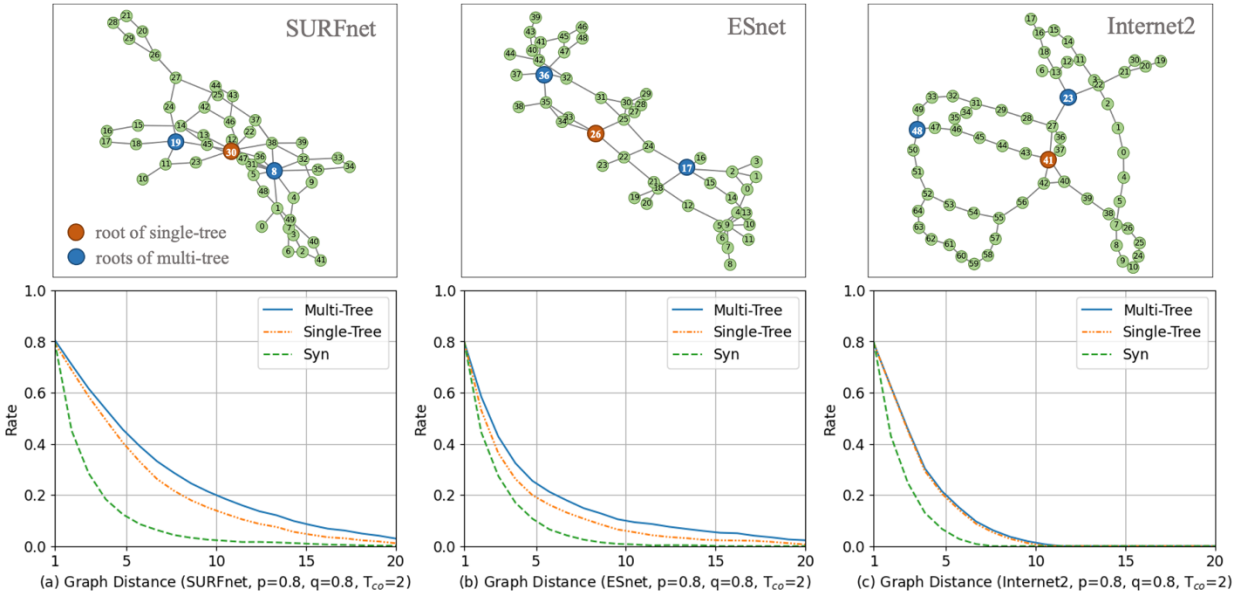


Figure 3.3. *Simulation Results for Realistic Topologies: SURFnet, ESnet and Internet2.*

SURFnet has nodes with strong connections, as indicated by their higher number of adjacent edges (e.g., Node 8 in Figure 3.3(a)), making them suitable candidates for root nodes. This suggests that SURFnet exhibits areas of high network density, resulting in superior multi-tree performance compared to the other two topologies. As shown in Figure 3.3(b)., ESnet exhibits less densely distributed nodes and fewer highly connected central nodes than SURFnet. Consequently, ESnet

performs less efficiently than SURFnet but maintains similar relationships between the different routing approaches, as shown in Figure 3.3(b).

In the case of Internet 2, as shown in Figure 3.3(c), which includes long linear chains in its topology, DODAG behavior becomes less stable and inefficient. When two nodes are in the middle of a long linear chain, their likelihood of being included in the DODAG diminishes. Consequently, all three routing schemes exhibit suboptimal performance in this topology, with both multi-tree and single-tree schemes displaying overlapping performance. This instance is an example of a topology that lacks even connectivity and is not well-suited for multi-tree routing.

One potential solution to this is the inclusion of floating DODAGs. Floating DODAGs preserve detached branches as isolated DODAG segments when a direct-link entanglement is lost. During this period, the severed node temporarily serves as a provisional root for the subset until it successfully reconnects with the primary DODAG. Upon reconnection, the isolated segment reintegrates into the primary DODAG. It is important to note that only the provisional root node of a floating tree can join a stationary DODAG with a real root node. Further simulations involving floating trees are part of future work.

Overall, in realistic topologies, multi-tree routing shows a modest performance improvement compared to single-tree routing. This discrepancy can be attributed to the relatively modest sizes of the research networks, whereas the multi-tree approach is designed for large networks. In other words, the nodes in these realistic networks generally have shorter inter-node distances on average than those in the generated larger topologies. Nevertheless, the overarching trend suggests that asynchronous routing, in both single or multiple tree versions, outperforms synchronous routing in various simulated and realistic topologies.

3.4 Summary

Building upon the foundation laid by our previous work, which focused on asynchronous routing with local knowledge to improve E2E entanglement rates, this study proposes a multi-tree approach using a DODAG framework, addressing the challenge of long-distance entanglement distribution. Our simulations demonstrate improvements in entanglement rates across various network topologies, including grids and barbells, as well as real-world testbeds like ESnet and Internet2. These results highlight the efficiency and advantages of asynchronous routing schemes. This reinforces the potential of asynchronous strategies in optimizing future quantum networks.

Chapter 4: Resource Management in DQC

Consider a quantum data center housing multiple QPUs distributed across different racks, where executing remote gates in quantum circuits requires entanglement between separate QPUs. The generation of this E2E entanglement often results in congestion and resource contention. In this chapter, we propose a resource management framework that maximizes fidelity-guaranteed throughput while assuring dependencies. The problem is first formulated as a MILP optimization model to serve as a benchmark. We then introduce efficient heuristic scheduling algorithms that perform comparably to the optimization solver. While there is a tradeoff between execution time and throughput, simulations show that our heuristics closely approximate the solver's results in most scenarios. This indicates that simple heuristics are sufficient in current DQC settings, offering practicality for managing the potentially large number of remote gates in a quantum circuit. Such efficiency gains can significantly enhance overall performance in DQC systems.

4.1 Introduction

Recent advances in quantum computing, marked by the introduction of increasingly powerful quantum machines and advancing communication experiments, signify substantial progress in the field [1], [35], [36], [121]. As quantum computing continues to evolve, there is a growing demand to interconnect these systems, especially within a data center connected by optical switches (analogous to traditional circuit-switched networks). This can be achieved through DQC [188], [189], [190], [191], in which individual quantum processors, each with limited capacity, work together to address computational tasks that exceed the capabilities of a single device. Unlike classical distributed computing, where adding more processors leads to a linear increase in

computing power, the integration of additional quantum processors can result in an exponential enhancement of quantum computational capacity [14].

Quantum computers can be interconnected using a hybrid approach combining quantum and classical techniques [15], [17], [21], [26]. Hybrid quantum-classical networks often rely on a central controller to handle orchestration tasks like scheduling and routing, a practice that extends to DQC. However, scaling quantum resources within data centers presents unique challenges due to the inherent complexities of quantum technology, including decoherence, probabilistic operations, and unreliable entanglement generation [115]. Addressing these challenges calls for innovative strategies specifically designed for DQC, aligned with the shift from monolithic to distributed architectures. Enhancing performance necessitates unique resource management and optimization techniques suited to the quantum domain.

In the current NISQ era [10], executing a complex quantum algorithm in DQC involves partitioning the algorithm into smaller components (i.e., dividing the quantum circuit into sub-circuits) and allocating them to specific QPUs. Each sub-circuit comprises a set of quantum gates, some of which can be executed locally within a QPU, while others require resources and collaboration across QPUs, i.e., inter-processor operations, specifically referred to as remote gates. A primary goal in quantum algorithm partitioning is to reduce the number of remote gates, as their execution is significantly more resource-intensive compared to local gates. However, achieving this is highly challenging and inherently constrained [192], [193]. Therefore, minimizing the overhead associated with remote gate execution—such as employing resource management to enable their parallel execution—is equally important. Resource management strategies from classical networks cannot be directly applied to quantum networks due to the above differences [161].

A key method for executing remote gates between two QPUs is by leveraging a pre-established entangled pair shared between them [14]. Thus, an overlay network for entanglement distribution is crucial for enabling remote operations in DQC [116], [129]. This overlay network facilitates the distribution of E2E entanglement between arbitrary nodes in the network. The end nodes then utilize this E2E entanglement as a communication resource to perform remote operations. For entanglement distribution to function, each node in the network must communicate to coordinate the implementation of local quantum operations, also known as LOCC [87]. Once paths between end nodes are established, intermediate nodes (i.e., quantum repeaters) execute entanglement swapping and purification [52], [170] to produce E2E entanglement of sufficient quality. The key challenge lies in effectively selecting paths for concurrent remote gates, which require specific QPUs for execution, to achieve optimal performance. This involves maximizing the availability of resources, such as E2E entanglement with desired quality, for a batch of remote gates within a given time frame.

Consider a quantum data center with multiple QPUs distributed across various racks tasked with executing one or more complex quantum circuits in a distributed manner. At any given time, numerous requests arise for resources to execute remote gates in the quantum circuit, each requiring E2E entanglement between QPUs. Ideally, maximizing the concurrent execution of remote gates is desired. However, as the number of requests increases, congestion and resource contention become inevitable. For instance, multiple remote gates may rely on a shared link, but the resources available between the QPUs connected by that link may be insufficient to meet all demands.

To address this, we propose a resource management framework for quantum data centers that prioritizes gate dependencies while maximizing quality-assured resources for remote gates. The

problem is first formulated as a MILP optimization model, serving as a benchmark. We then introduce a set of efficient heuristic scheduling algorithms, such as prioritizing requests with minimal resource demands. While there is a tradeoff between execution time and request satisfaction, simulations show that the proposed heuristics closely approximate the solver's results in most scenarios.

This demonstrates that simple heuristic algorithms are sufficient for current DQC settings. Unlike sophisticated methods such as reinforcement learning or optimization solvers used in existing works, a straightforward heuristic can achieve comparable performance with significantly faster execution times. Even in harsh conditions, the results are acceptable. That is, an efficient heuristic can offer a more practical solution for managing the potentially large number of remote gates in a quantum circuit. These efficiency improvements can greatly enhance overall performance, especially for large quantum computations involving thousands or even millions of sub-circuits. Saving time on processing one batch of remote gates—a set of sub-circuits—can translate into substantial time savings over the course of the entire computation.

With that, the contributions of this chapter can be summarized as follows:

1. We propose a resource management framework for quantum data centers that abstractly models the components of a partitioned quantum circuit executed in a DQC environment.
2. We formulate the resource management problem as a MILP model, which serves as a reference for performance evaluation.
3. We introduce a set of efficient heuristic algorithms that deliver performance comparable to MILP while achieving significantly faster execution.

4. We assert that straightforward heuristic scheduling is both sufficient and practical for current DQC settings, outperforming the complexity of sophisticated methods employed in existing works.

To fully understand the content of this chapter, readers are strongly encouraged to review Section 1.3 and Subsection 1.5.2 beforehand.

4.2 Topologies for DQC

In this section, we introduce the network topologies identified for connecting QPUs in quantum data centers. A straightforward approach is to adopt topologies inspired by those used in early classical data centers [194], [195], and analyzed in the context of quantum networks [196], [197]. One such consideration is the widely discussed entanglement switch [198], [199], designed for dynamically switchable entanglement networks, which in its early stages is expected to resemble a Clos network—a multistage circuit-switching network [200]. By adding stages, a Clos network reduces the number of cross points needed to construct a large switch, e.g., a hub. In this context, the distribution of E2E entanglement is analogous to circuit switching in classical networks, where a dedicated communication path is established between end nodes for the duration of the connection. Based on these principles, we explore two types of initial topologies for quantum data centers: DCell and BCube [201].

DCell and BCube were both originally developed to address the limitations of traditional hierarchical data center networks, offering scalability, fault tolerance, and manageability. These topologies are well-suited for quantum data centers, where efficient interconnectivity between QPUs is critical for DQC. DCell is defined recursively: a DCell at one level (referred to as Cell 0) acts as a building block for higher levels (Cell 1, Cell 2, etc.). For example, the topology shown in

Figure 4.1(a) is a 5-cell DCell topology. Note that Quantum Hubs (i.e., Q-Hubs) in a DCell mainly serve as internal optical switches within a cell while facilitating intra-cell switching. BCube, on the other hand, employs a multi-level recursive design inspired by the hypercube. In this topology, QPUs are interconnected through multiple layers of switches, creating a structure that supports robust connection paths and efficient fault recovery. As shown in Figure 4.1(b), a 2-level BCube topology showcases the layered interconnections between nodes, enabling greater network capacity and manageability compared to traditional designs.

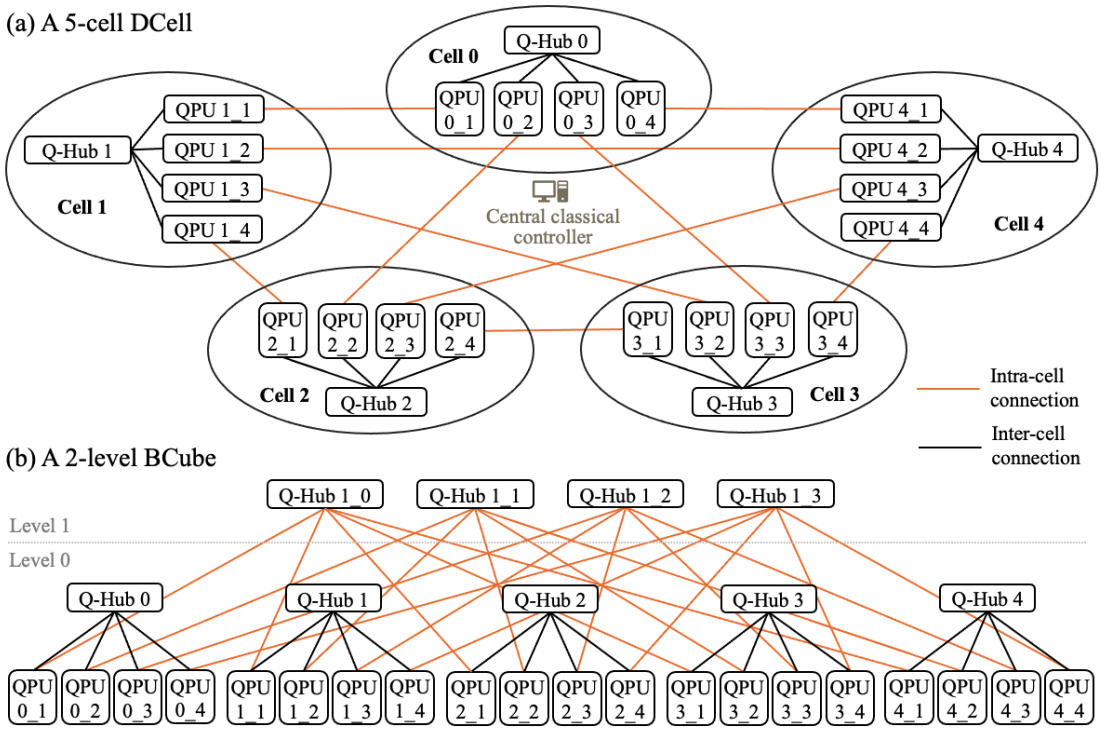


Figure 4.1. *DCell and BCube Topologies for Quantum Data Centers.*

In both topologies, a central controller oversees network orchestration, handling tasks such as routing and scheduling for entanglement distribution. DCell and BCube topologies are ideal candidates for the inter-rack connections. Nonetheless, we envision a hybrid approach in near-term quantum data centers: using DCell and BCube for inter-rack connectivity and employing simpler topologies, such as star or tree configurations, for intra-rack connections. In this chapter, we focus

on the inter-rack topologies—DCell and BCube—due to their suitability for managing distributed quantum operations. Future work will explore the integration of intra-rack connections to complement these designs.

Similar to previous chapters, we refer to the topologies formed by physical quantum channels, such as DCell and BCube, as physical topologies. To enable the distribution of E2E entanglement, a pre-establishment phase is required, during which each node generates a direct-link entangled pair with its neighbors. In contrast to the physical topology, we define the topology formed by entanglement links as the instant topology, reflecting its dynamic nature. Ideally, the instant topology would match the physical topology if direct-link entanglement generation were fully successful. However, in practice, the instant topology is often a subset of the physical topology, with link attributes reflecting varying entanglement link fidelities and capacities.

4.3 Resource Management Framework for DQC

Quantum data centers face unique constraints that do not apply to classical data centers. For instance, although a qubit can be transmitted to a remote node by encoding it into a photon's degree of freedom and sending it through a fiber link or free space, any loss of the photon due to attenuation or decoherence makes the information it carries irretrievable, as it cannot be copied or retransmitted [46]. Entanglement generation is inherently probabilistic, resulting in dynamic connections and constantly shifting network topologies. Furthermore, entangled states are highly unstable, with lifetimes ranging from milliseconds to seconds [202]. These challenges necessitate the development of adaptive routing and scheduling algorithms to ensure efficient and reliable operation [26].

Therefore, a specialized resource management approach that accounts for quantum data centers' unique quantum properties, components, and architectures is crucial. In this context, we foresee two primary types of resource contention: one arising from the execution of dense sets of remote two-qubit gates within a quantum circuit (i.e., single-tenancy), and another stemming from running multiple quantum circuits concurrently on the same network (i.e., multi-tenancy). To effectively address resource contention in these scenarios, the approach must maximize the number of satisfied requests—referred to as network throughput—at any given time while ensuring fairness by proper request sequencing and satisfying request priority. In this chapter, we focus on addressing the single-tenancy scenario, but the approach can be adapted to support multi-tenancy.

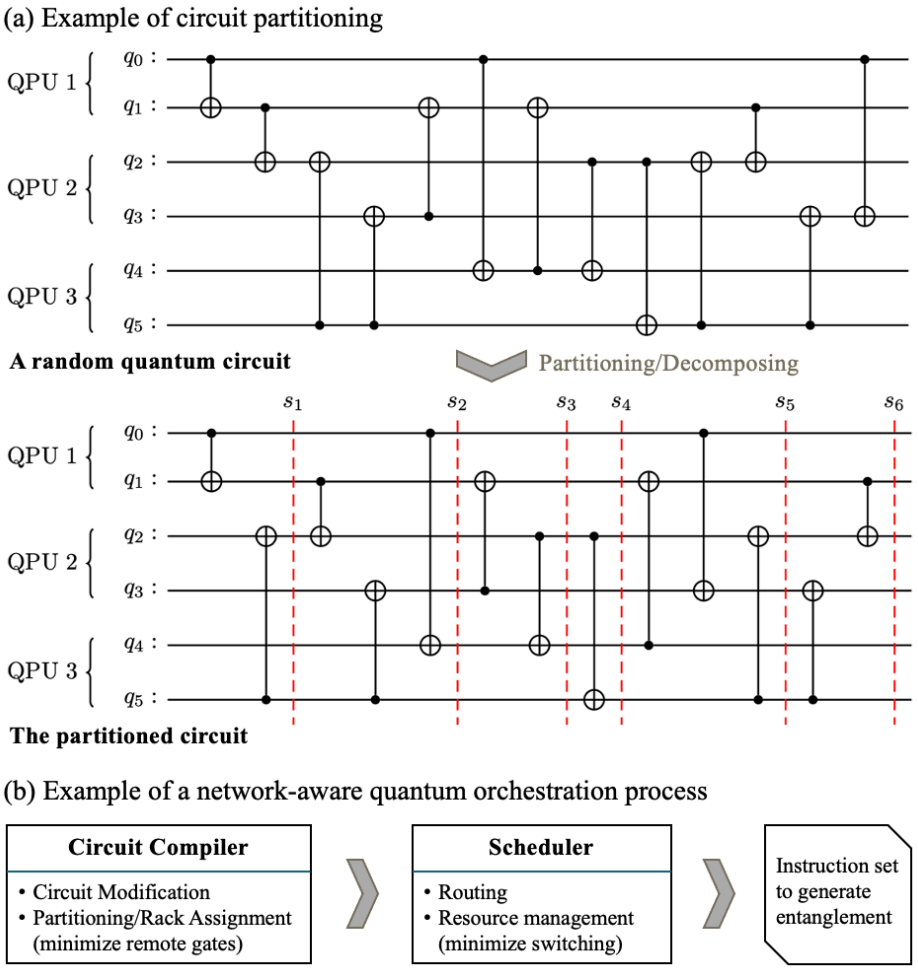


Figure 4.2. *Example of (a) Circuit Partition and (b) Network-aware Quantum Orchestration.*

In a single-tenancy quantum data center, a quantum circuit is partitioned by a circuit compiler such that the order of remote gates within a sub-circuit is arbitrary, while the order of sub-circuits remains essential, as illustrated in Figure 4.2(a). The partitioned sub-circuits are then processed by a scheduler to generate an instruction set for the generation of E2E entanglement, as shown in Figure 4.2(b). Multiple QPUs may be located in different network cells within the quantum data center. At any given time, the scheduler handles numerous sub-circuits containing remote gates that request E2E entangled pairs between QPUs. Many of these requests may rely on the same link to establish the required entangled pairs, but the available resources on each link are limited. Based on these configurations, the resource management process in a quantum data center is defined as follows.

The symbols and notations used in this chapter are listed in Table 4.1. As indicated, $G(V, E)$ represents the physical topology of a quantum data center, where each node $v \in V$ corresponds to a QPU, and each edge $e \in E$ represents a link connecting two QPUs. Each link e has a maximum capacity of C_e, representing the maximum number of entanglement generations it can support between adjacent nodes. At any given time, the active link capacity, denoted as $c_e \leq C_e$, refers to the number of entangled pairs that have been successfully generated on the link e. Each direct-link entangled pair initially has a fidelity f_e. Note that $G(V, E)$, with zero entanglement edges removed, i.e., $c_e = 0,$, is called the instant topology.

At a given time, there exists a batch of requests from a group of remote gates, R, requiring E2E entangled pairs between distant QPUs for execution in a specific sequence. These remote gates can be from one or more sub-circuits within a single quantum circuit or across multiple quantum circuits. Each remote gate request r is characterized by two required QPUs, a duration d_r to complete, a weight w_r, a maximum acceptable path length h_r^{max} (i.e., the maximum number of

hops), a minimum fidelity f_r and a dependency set, D_r, containing the requests that need to be finished before it. Multiple paths are identified for each request r, denoted by P_r, with each path ρ having a hop length h_r^ρ. For simplicity, it is assumed that all entangle pairs share a uniform coherence time t^{co}, meaning all requests must be completed within this timeframe.

Table 4.1. Symbol Table

Notation	Description
$G(V,E)$	The network topology in a quantum data center, with nodes represented by set V and links by set E.
c_e	The capacity of link $e \in E$: the number of entangled pairs on link e.
f_e	Fidelity of a direct-link entanglement generated by link e.
R	Set of requests from a group of remote gates.
s_r	Start time for request $r \in R$.
d_r	Duration to finish the request r.
w_r	Weight of request r.
D_r	Dependency set for request r. The request r can only begin after all the requests in D_r have been completed.
h_r^{max}	Maximum number of hops for each path associated with request r to satisfy the minimum entanglement rate for request r.
f_r^{min}	Fidelity threshold for request r.
P_r	A set of paths that connect QPUs required for the request r. The paths are arranged in ascending order by their length.
f_r^ρ	Fidelity that can be achieved by using path $\rho \in \mathrm{P}_r$ for request r.
f_e^ρ	Distilled link fidelity on link e for path ρ.
h_r^ρ	Length of path ρ (number of hops) for request r.
c_r^ρ	The capacity required for purification and meeting the fidelity requirement of request r over path ρ.
ξ_r^ρ	Entanglement rate for request r along path ρ.
p	Direct-link entanglement generation success probability.
q	Entanglement swapping success probability.
t^{co}	The coherence time of the Bell pairs in a quantum data center, which serves as the deadline for completing all requests in R.

The management process for requests from remote gates in a quantum data center is summarized as follows:

1. Each node generates entangled pairs with its neighbors, forming an instant topology $G(V, E)$ in which each link's capacity is determined by its entanglement generation outcome. Each direct-link entanglement within a link is generated with a probability of p.
2. The network receives a batch of requests, R, from a group of sub-circuits, which contain remote gates.
3. A set of paths, P_r, are identified for each request r. The number of paths $|\mathrm{P}_r|$ for each request should be limited to $|\mathrm{P}_r|^{max}$ to ensure fairness.
4. The best path selection and start time for each request are determined.
5. Go to 1 and repeat the process for next batch of requests.

The challenge in this process involves selecting the optimal paths and arranging the correct sequence of requests in step 4.

4.4 Optimization and Heuristics

As previously discussed, quantum data centers face unique constraints that are not present in classical data centers, requiring specialized strategies. In this section, we define the problem and present the proposed models and algorithms for resource management in DQC within quantum data centers.

4.4.1 Optimization by Mixed Integer Linear Programming

To maximize the number of satisfied weighted requests for fidelity-guaranteed E2E entangled pairs—referred to as weighted throughput—within a single time unit (i.e., one coherence time,

before any direct-link entangled pairs decohere and the topology changes), while preserving request dependency relationships and ensuring fairness, we define the resource allocation problem as follows:

$$\text{maximize} \sum_{\forall r \in R} \sum_{\forall \rho \in \mathrm{P}_r} x_r^\rho \cdot w_r \cdot \xi_r^\rho \qquad (4.1)$$

subject to

$$\sum_{\forall r \in R} \sum_{\forall \rho \in \mathrm{P}_r : e \in \rho} x_r^\rho \cdot c_r^\rho \leq c_e, \forall e \in E \qquad (4.2)$$

$$s_i + d_i \leq s_r, \forall r \in R, \forall i \in D_r \qquad (4.3)$$

$$h_r^\rho \leq h_r^{max}, \forall \rho \in \mathrm{P}_r, \forall r \in R \qquad (4.4)$$

$$f_r^\rho \geq f_r^{min}, \forall \rho \in \mathrm{P}_r, \forall r \in R \qquad (4.5)$$

$$\max_{\forall r \in R}(s_r + d_r) \leq t^{co} \qquad (4.6)$$

The objective of the optimization problem in (4.1) is to maximize the weighted network throughput for a given set of requests, which need to be satisfied within one coherence time. The weighted throughput ensures that requests with higher weights are executed with choices of shorter path or more paths (i.e., higher rate for higher weight) within the constraints. The decision variables are $\{x_r^\rho, s_r\}$ where $x_r^\rho \in \{0,1\}$ denotes the selection of path ρ for request r (with 1 indicating ρ is selected, otherwise, 0), and s_r denotes the start time of the request r, which ensures the predefined request precedence. w_r indicates the weight of request r. ξ_r^ρ denotes the resulting E2E entanglement rate for request r when using path ρ and can be computed by $\xi_r^\rho = q^{h_r^\rho - 1}$ (i.e., the success probability of a sequence of swapping operations).

The constraint in (4.2) ensures the selected paths together do not exceed the capacity of any link in any path. c_r^ρ denotes the capacity path ρ required for purification to meet the fidelity requirement of the request r (i.e., each link that path ρ has needs to have available capacity greater than c_r^ρ).

Before computing c_r^ρ, we need to know the minimum fidelity each link of path ρ needs to have, to achieve

f_r^{min}, given the number of swapping needed. That is, we need to consider fidelity changes after a number of swapping (fidelity as a function of path length). For a path with a length of h_r^ρ and distilled link fidelity for path ρ, f_e^ρ, the resulting fidelity of the E2E Bell pair path ρ can achieve (after undergoing $l_r^\rho - 1$ nested swap operations) can be expressed as [5]:

$$f_r^\rho = \frac{1}{4} + \frac{3}{4}\left(\frac{\phi(4\eta^2 - 1)}{3}\right)^{l_r^\rho - 2}\left(\frac{4f_e^\rho - 1}{3}\right)^{l_r^\rho - 1} \qquad (4.7)$$

where ϕ is the two-qubit gate fidelity and η is the measurement fidelity of the swapping operation. To compute f_e^ρ, we need to consider the purification relationship between resulting fidelity and direct-link Bell pair fidelity f_e. Without loss of generality, by considering bit flip errors only, the resulting fidelity after purification using two Bell pairs can be computed by [4]:

$$F(f_1, f_2) = \frac{f_1 f_2}{f_1 f_2 + (1 - f_1)(1 - f_2)} \qquad (4.8)$$

If we consider the uniform fidelity of Bell pairs on the same edge, then $f_1 = f_2$, and the formula becomes $F(f) = \frac{f^2}{2f^2 - 2f + 1}$. In principle, this process can be applied recursively to achieve arbitrarily high fidelities by consuming a number of Bell pairs. Using a recursive approach can determine the round of purifications (or the number of Bell pairs) needed to achieve the target

fidelity, which thus compute c_r^ρ. For effective purposes, an iterative approximation can be applied. More details can be found in Appendix E.

Let us assume we use the same amount of direct-link Bell pairs in each link of path ρ. That is, $c_r^\rho = n + 1$ where n is the round of recursive purification conducted. Then, we have the same distilled link fidelity on each link for path ρ, i.e., f_e^ρ. Then, with (4.8), we have:

$$f_e^\rho = \begin{cases} F_n = F_{n-1}(F_{n-2}, f_e), & n \geq 2 \\ F_1 = F(f_e, f_e), & n = 1 \\ F_0 = f_e, & n = 0 \end{cases} \quad (4.9)$$

where F_{n-1} is a recursive form based on (4.8) when n>=2. F_1 is the same as (4.8) and $F_0 = f_e$. Let $f_r^\rho = f_r^{min}$ and solve for n using (4.7) and (4.9), we get $c_r^\rho = n + 1$.

The constraint in (4.3) ensures that requests must respect predefined sequences and dependencies. If request i precedes request r (i.e., $i \in D_r$), the completion time of request i (i.e., $s_i + d_i$) must be less than or equal to the start time of request r, s_r. The constraint in (4.4) ensures the practicality of each path by setting a maximum path length for each path associated with request r to satisfy the minimum entanglement rate for request r. The constraint in (4.5) ensures the fidelity achieved by using path ρ for request r is greater than the fidelity threshold for request r. The constraint in (4.6) ensures the latest completion time of all requests is less than the coherence time of the Bell pairs in a quantum data center (i.e., the deadline for completing all requests of the current batch).

It is worth noting that if the batch of requests is time-sensitive, where we want to minimize the maximum completion time (makespan) while maximizing throughput, the objective function can be changed to reflect the dual objectives. These two objectives often conflict — maximizing throughput could increase execution time due to resource saturation, and minimizing execution

time might reduce the opportunities to maximize throughput. Thus, if makespan matters so much that throughput can be compromised (or balanced), it can be modeled as a dual-objective optimization problem that simultaneously maximizes throughput and minimizes execution time. The second objective can be expressed as

$$\text{minimize } O_2 = \text{minimize} \max_{\forall r \in R}(s_r + d_r) \qquad (4.10)$$

Given the dual objectives, we can model them as a weighted sum objective:

$$\text{maximize } \alpha O_1 - \beta O_2 \qquad (4.11)$$

where O_1 is the first objective function $O_1 = \sum_{\forall r \in R} \sum_{\forall \rho \in \mathrm{P}_r} x_r^{\rho} \cdot w_r \cdot \xi_r^{\rho}$ in (4.1). α and β are weights indicating the importance of maximizing throughput and minimizing execution time, respectively.

Alternatively, we can use a ε-constraint model where one objective is optimized while the other is included as a constraint to be satisfied:

$$\text{maximize } O_1 \;\; s.t. \;\; O_2 \leq \varepsilon \qquad (4.12)$$

or vice versa, where ε is a threshold that you are willing to accept for the constrained objective.

Nonetheless, in the rest of this chapter, for simplicity, we consider throughput the sole objective as shown in (4.1) and include global makespan as a constraint to be satisfied. The threshold for the constrained makespan is that it is always less than t^{co}, which makes the allocation feasible.

4.4.2 Resource Optimization with Non-overlapping Constraint

In certain scenarios within a quantum data center, it is essential to ensure that no two requests are scheduled to overlap in time on the same machine. For example, consider a scenario where a quantum data center is tasked with processing several remote gates that require the utilization of a specialized QPU. Each request needs to execute quantum algorithms that involve complex calculations such as factoring large numbers, simulating molecular interactions, or optimizing financial portfolios. These tasks are computationally intensive and sensitive to the precise control of quantum states, necessitating dedicated access to the QPU resources.

If two requests, i and r, are assigned to the same QPUs and their execution times overlap, it could lead to quantum state interference, erroneous computation results, or degradation in system performance. For instance, if the remote gates of request i involves preparing and measuring quantum states that are susceptible to decoherence over time and gates in request r concurrently starts an operation that unintentionally impacts these states due to shared resource contention, both requests might fail to achieve their goals. To manage this, we can utilize a non-overlapping constraint expressed as below, which can be added to (4.1)'s constraint set when necessary:

$$s_i + d_i \leq s_r \;\; or \;\; s_r + d_r \leq s_i, \forall i, r \in R, i \neq r \quad if \; \{a_i, b_i\} \cap \{a_r, b_r\} \neq \emptyset \tag{4.13}$$

where a_i and b_i are the source node and the destination node of request i, respectively. This constraint ensures that for each pair of requests assigned to the same QPUs, one of the following conditions must hold: either the completion time of the first request must be less than or equal to the start time of the second request or the completion time of the second request must be less than or equal to the start time of the first request. The "if" condition checks if there is any common

node between the sets of source and destination nodes of requests i and r. This captures any overlap among the source and destination nodes of the two requests.

To model the logical operations in (4.13) so as to implement this constraint effectively in a MILP model, we can use a big-M model to linearize the disjunction [203]. We can introduce a binary variable indicating if request i is scheduled before request r, $y_{ir} \in \{0,1\}$. It equals to 1 when request i is in the dependency set of request r. Then, the constraints can be expressed as:

$$s_i + d_i \leq s_r + M(1 - y_{ir}), s_r + d_r \leq s_i + My_{ir} \quad \forall i, r \in R, i \neq r \tag{4.14}$$

where M is a sufficiently large number that makes the constraint non-binding when not required.

4.4.3 Scheduling by Heuristics

MILP may be appropriate if the primary goal is to determine an optimal allocation of paths and resources for a static set of requests and constraints without factoring in the time required to compute the solution. It is particularly effective when relationships can be linearized, and the environment remains stable. However, in a quantum data center, the instant topology changes dynamically with each coherence time, and incoming request batches are inherently dynamic. More critically, in the current NISQ era, the coherence time is very short, necessitating fast solutions to address the current instant topology before it changes. To address these challenges, we propose a set of efficient heuristic scheduling algorithms as follows:

We first propose the Least Resource First (LRF) scheduling algorithm, which prioritizes paths that consume the least resources. It selects paths that leave the maximum resources available post-allocation while respecting the dependency relationships between requests. Specifically, it evaluates

multiple paths for each request to identify the one that minimizes resource usage, thereby reducing the impact on the overall system capacity. LRF prevents over-commitment to resource-intensive requests, reducing the risk of scheduling bottlenecks. Its solution space is relatively small, making it exceptionally efficient.

To enhance performance by expanding the solution space, we extend LRF to propose the Group LRF (GLRF) algorithm. GLRF processes a group of requests (e.g., two requests per group) simultaneously, selecting paths collectively that leave the most resources free after allocation. This broader exploration of the solution space provides better optimization compared to processing requests individually with LRF. For instance, when evaluating solutions for a batch containing k requests, GLRF considers $C(k, 2)$ possible combinations, while LRF only evaluates k. Here, $C(k, 2)$ represents the "given k choose 2" combinatorial. By assessing the collective impact of multiple requests on resource allocation, GLRF identifies more effective solutions. Prior to path selection, requests are sorted based on their precedence sets using a directed acyclic graph (DAG) by topological sorting.

Considering the importance of weighted throughput (not just throughput), we further introduce the Weighted GLRF (WGLRF) algorithm, which incorporates request weights into GLRF. WGLRF prioritizes groups of requests with higher total weights, where group weight is the sum of the weights of all requests in the group. This improves performance by emphasizing high-priority requests. Like GLRF, WGLRF explores group-based solutions but adds a weight-based prioritization step. The three algorithms—LRF, GLRF, and WGLRF—are computationally efficient, with WGLRF closely approximating optimization results in many scenarios.

Since WGLRF encompasses the functionality of both GLRF and LRF, we present only the pseudocode for WGLRF with Algorithm 4.1 in Figure 4.3. For simplicity, the pseudocode shows the assignment of a single path to each request. However, it can be easily adapted to assign multiple paths to a single request when additional link capacity is available after all requests in the batch have been assigned one path. Weighted topological sorting is used to order non-dependent vertices by weight during topological sorting [204].

Algorithm 1. Weighted Group Least Resource First (WGLRF)

Input: Graph G, Requests R and Number of requests per group n
Output: Assignment of paths to requests with start times

```
assignment ← empty dictionary
# Sort requests by weighted topological order
sorted_requests ← weighted_topological_sort_requests(R)
for i from 0 to len(sorted_requests) step n do
    group_requests ← sorted_requests[i:i + n]
    for each request r in group_requests do
        earliest_start ← get_earliest_start_time(assignment)
        min_resource_usage ← infinite
        for each path ρ in r['paths'] do
            resource_usage ← get_required_bell_pairs(G[u][v]['fidelity'],
                r['min_fidelity'], len(ρ))
            # Check path feasibility and path optimality
            if G[u][v]['capacity'] ≥ resource_usage and
                resource_usage < min_resource_usage then
                # Assign path to request with earliest start time
                append (r, ρ, earliest_start) to assignment
                G[u][v]['capacity'] −= resource_usage
return assignment
```

Figure 4.3. *Algorithm 4.1. Weighted Group Least Resource First (WGLRF).*

WGLRF naturally promotes a more balanced distribution of load across system resources, preventing overuse of any particular link by focusing on minimizing resource usage. In multi-request or multi-tenant environments, this ensures that no single request monopolizes resources, improving

overall fairness and request satisfaction. The WGLRF strategy is conceptually straightforward and can be implemented using simple algorithms to evaluate resource usage for each path.

Note that, in all methods, paths are evaluated to ensure they meet the fidelity requirements of a request before being assigned. Paths that cannot achieve the required fidelity through entanglement purification are excluded from consideration. Additionally, requests with unresolved dependencies are temporarily deferred until their dependencies are resolved, and requests that cannot be satisfied within the current timeframe are carried over to the next batch.

4.5 Evaluation

To evaluate the performance of our proposed algorithms, we use the simple First Come First Served (FCFS) algorithm as a lower-bound benchmark and the CPLEX MILP solver [205] for optimization problem (4.1) as an upper bound. FCFS processes requests based on their dependencies and weights, starting with those that have none and progressing to those with increasing levels of dependency.

4.5.1 Time Complexity

Branch-and-bound is widely used in integer programming to solve MILP problems, including CPLEX. The method systematically divides the problem (branching) and uses solution bounds of subproblems (bounding) to gradually narrow down the search for the optimal solution. In our case, each branching point (or decision) in the solution space can potentially result in the generation of two new subproblems or branches. This bifurcation occurs because each variable at a decision point can lead to two distinct paths: one where the variable is included in the solution and one where it is not. As a result, the number of subproblems—or nodes in the decision tree—can double with each additional variable. This exponential growth in the number of nodes that need to be evaluated reflects the inherent complexity of the problem.

Given that n potential paths are identified for the requests in a batch, with each path requiring a binary decision (branch), the total number of possible combinations of these decisions is 2^n. This corresponds to the total number of leaf nodes in a fully expanded binary tree, with each node representing a subproblem that may need to be resolved in the worst case. Consequently, the time complexity of a branch-and-bound-based solver is $O(2^n)$. The complexity of handling dependency constraints is negligible compared to 2^n and thus is omitted. Let k represent the number of requests and m denote the constant number of potential paths per request. Since the total number of paths $n = k * m$, the complexity can be represented as $O(2^k)$.

In FCFS, requests are processed sequentially based on their order of arrival and the resolution of their dependencies. This can be achieved using topological sorting, which has a time complexity of $O(k + \delta)$, where δ represents the number of directed edges in the dependency graph. Each edge signifies a precedence constraint, indicating that one request must be completed before another can begin.

LRF not only processes requests based on their dependency resolution but also evaluates which path for each request minimizes resource usage. This involves a comparative analysis of all paths per request to determine the one that leaves the most resources free. It involves determining the least-resource path among all paths of a request, together with dependency checks. This process sequentially evaluates each path until a valid one is found. Assuming a uniform distribution of paths across requests, the complexity is $O(k * m + \delta)$, thus $O(k + \delta)$.

In GLRF, requests are grouped based on criteria like minimum dependencies. For each group, GLRF evaluates multiple paths for each request in the group to determine which combination of paths across all requests in the group minimizes the total resource usage. First of all, we still need

topological sorting which is $O(k+\delta)$. Then, for fixed group sizes, resource evaluation for each group has a complexity related to the combinations of requests within each group. For each group, $C(k,g) = O(k^g)$ combinations may be considered, where g is the number of requests within the group. Including the sorting and grouping, the complexity can be represented as $O(k+\delta+k^g * m * k/g)$. Since m and g are constant, the complexity is $O(k+\delta+k^{g+1})$ and thus $O(k^{g+1})$.

WGLRF is similar to GLRF, but adds a preliminary step of weighting groups, which might involve additional complexity depending on how weights are calculated, and groups are formed. This can add an $O(klogk)$ component for sorting by weight before topological sorting. Thus, the complexity of WGLRF is $O(klogk+k^{g+1})$. Since $O(klogk)$ grows slower than $O(k^{g+1})$ for any $g \geq 1$, it can be simplified as $O(k^{g+1})$.

In short, as expected, the optimization solver's complexity is exponential for the branch-and-bound method to explore the solution set. All other algorithms are polynomial.

4.5.2 Simulation Settings

To evaluate the performance and efficiency of various heuristic and optimization approaches in the aforementioned quantum data center configuration, we conduct simulations using NetworkX [206] to model the dynamic network graph. A uniform-distribution-based random number generator is used to generate random circuits that produce weighted requests with random dependencies. For simplicity, the direct-link entanglement generation success probability is set to a constant value of 0.8, introducing variability in link capacities during the simulation.

We simulate the 5-cell DCell and 2-level BCube topologies separately to assess the methods' compatibility with different network structures. Both topologies are configured with the same number of end nodes (QPUs), complemented by quantum hubs positioned to facilitate connectivity.

Remote gate pairs are generated randomly among QPUs, with switches and hubs excluded as end nodes. The requests generated from these remote gates correspond to E2E entangled pairs required between the selected QPU pairs. Network graphs are constructed with edge attributes, including maximum capacity and initial fidelity of the direct-link entanglement, while vertices are categorized as either QPUs or switches.

A random partitioned quantum circuit, such as the one shown at the bottom of Figure 4.2(a), is generated to form the basis of the simulation. Requests are grouped into batches corresponding to the partitioned circuit: gates in a sub-circuit on the right depend on the completion of all gates in sub-circuits to its left. Each gate generates a request, which includes two randomly selected QPUs requiring an E2E entangled pair and a randomly generated minimum fidelity requirement. For each request, five potential paths are identified using a k-shortest path algorithm [207]. The proposed algorithms are then implemented to determine the optimal combination of paths for all (or as many as possible) requests in a batch, considering the link capacity constraints and fidelity requirements.

The simulation iterates over a range of request sets, with each set varying by the number of requests, their complexity, and interdependencies. As discussed, instead of merely counting the total number of requests processed, the simulation evaluates the weighted throughput, which measures the total "value" of requests successfully scheduled and completed, reflecting the priority and importance of requests handled by the scheduler. Additionally, fairness in scheduling is assessed using Jain's Index [208], a widely used fairness metric that measures how evenly resources and scheduling opportunities are distributed among requests. Jain's Index is calculated as the square of the sum of allocated resources divided by the number of requests multiplied by the sum of the squares of the resources allocated to each request. A higher Jain's Index indicates a more equitable allocation, which is important in scenarios where requests vary significantly in priority and requirements.

4.5.3 Simulation Results

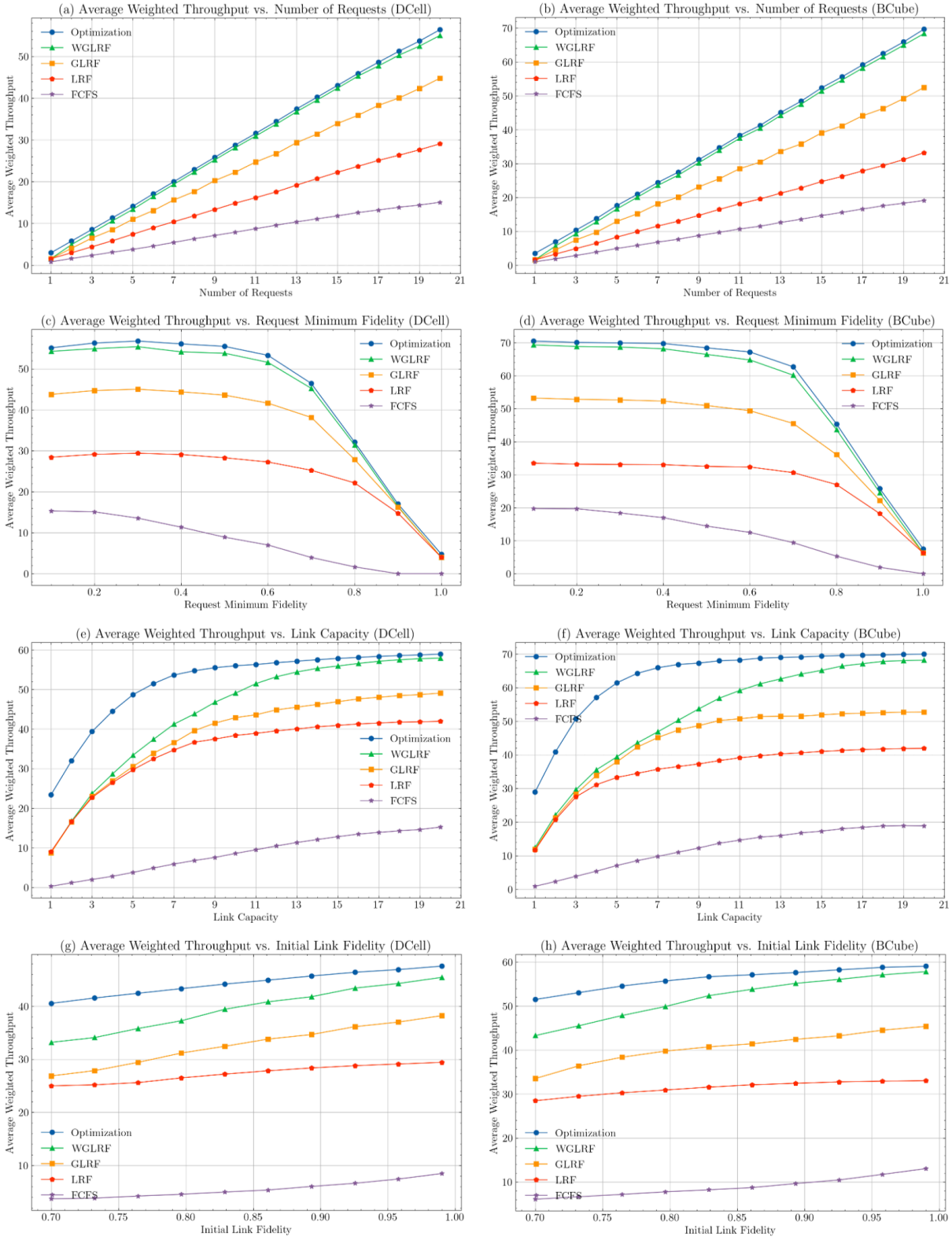


Figure 4.4. *Simulation Results of All Methods in DCell and BCube.*

Figure 4.4 presents the performance of all the discussed schedulers across multiple dimensions in both DCell and BCube topologies. Details are in the subsequent paragraphs. Note that for all simulations, when a factor is not explicitly mentioned or is not a variable on the x-axis, its default value is as follows: the link capacity is set to 20, meaning there are 20 entangled pairs available between two adjacent nodes; the minimum fidelity requirement for a request is randomly generated between 0.5 and 1; the initial fidelity of a direct-link entangled pair is 0.9; the maximum number of paths selected per request is 3; the maximum number of hops for a path is 10; and the number of requests per batch is 20.

As shown in Figure 4.4(a) and (b), for DCell and BCube respectively, the optimization solver consistently achieves the best performance. However, the weighted throughput achieved by WGLRF closely approximates that of the optimization solver, regardless of the number of requests in a batch. As anticipated, GLRF performs slightly worse than WGLRF but outperforms LRF. All these algorithms demonstrate better performance compared to FCFS.

As shown in Figure 4.4(c) and (d), the performance degrades significantly as the average minimum fidelity requirement for requests increases. This decline is attributed to fidelity loss during entanglement swapping, which necessitates using more entangled pairs to meet the fidelity requirement, thereby reducing available resources for other requests. Despite this, WGLRF continues to closely approximate the optimization solver in terms of weighted throughput, and the performance ranking of the methods remains unchanged.

Furthermore, as shown in Figure 4.4(e) and (f), while WGLRF cannot reach the solver performance when link capacity is low, its performance increasingly approaches that of the solver as link capacity grows. Similarly, the ranking of performance for the methods remains consistent. Lastly, as shown

in Figure 4.4(g) and (h), WGLRF is slightly inferior to the optimization solver at low initial fidelity for direct-link entanglement, but its performance becomes nearly optimal as the initial link fidelity improves.

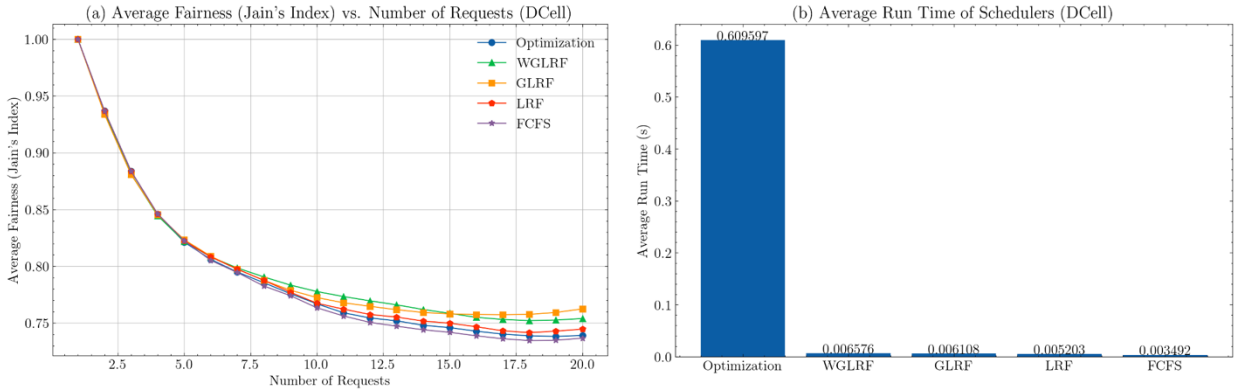


Figure 4.5. *Simulation Results of Fairness and Execution Time.*

Additionally, we conducted simulations to evaluate the fairness and runtime of the schedulers. As shown in Figure 4.5(a), all algorithms exhibit similar fairness, which is expected due to the presence of dependency relationships among requests. WGLRF and GLRF exhibit slightly improved fairness as the number of requests increases. In Figure 4.5(b), the execution times of WGLRF, GLRF, LRF, and FCFS are significantly lower than those of the optimization solver. Among these, runtime increases slightly from FCFS and LRF to WGLRF.

Overall, WGLRF closely approximates the results of the optimization solver when link capacity is relatively sufficient. As link capacity increases, the performance of WGLRF progressively aligns with that of the solver. Across varying minimum fidelity requirements, WGLRF remains close to optimal. At low initial link fidelity, WGLRF performs slightly below optimal, but it approaches optimal performance as the link fidelity increases. All algorithms demonstrate similar fairness, influenced by the dependency relationships. Additionally, all heuristic schedulers (i.e., WGLRF, GLRF, LRF) execute significantly faster than the optimization solver, with WGLRF maintaining

performance comparable to the optimization solver in most scenarios. Moreover, our simulations indicate that performance in BCube generally surpasses that in DCell.

These results demonstrate that, despite the tradeoff between execution time and weighted throughput, simple heuristic algorithms are sufficient for current DQC settings. Compared to optimization solvers, heuristics provide a practical solution for managing the large number of remote gates. Efficiency gains in processing batches of requests can substantially enhance overall performance in DQC systems. In the context of a quantum circuit containing millions of remote gates, such efficiency gains per batch become critical for executing the entire circuit efficiently.

4.6 Summary

With the limited computational power of a single quantum computer, DQC in quantum data centers offers immense potential to revolutionize fields such as cybersecurity, sensing, and pharmaceuticals, benefiting both industry and academia. However, quantum data centers may house a large number of QPUs, facilitating the execution of numerous remote gates in quantum circuits. Each remote gate demands entanglement between distinct QPUs with specific fidelity requirements. To address congestion and resource contention during the distribution of E2E entanglement for remote gates, we propose a resource management framework that captures the unique characteristics of quantum data centers. The problem is formulated as a MILP optimization model to serve as an upper-bound benchmark. We further introduce efficient scheduling that, in many scenarios, perform comparably to the solver. Although there is a tradeoff between execution time and throughput, simulations show that the heuristics closely approximate the solver's results in most cases. This suggests that simple heuristics are sufficient for current DQC settings, providing a practical solution for the potentially large number of remote gates in quantum circuits.

Chapter 5: Layer-wise Security Framework

This chapter explores the security challenges of quantum internet by analyzing the vulnerabilities and the corresponding mitigation strategies across different layers, including physical, link, network, and application layers. We assess the severity of potential attacks, evaluate the expected effectiveness of mitigation strategies, and identify vulnerabilities within diverse network configurations, integrating both classical and quantum approaches. It highlights the dynamic nature of these security issues and emphasizes the necessity for adaptive security measures. The findings underline the need for ongoing research into the security dimension of the quantum internet to ensure its robustness, encourage its adoption, and maximize its impact on society.

5.1 Introduction

Progress in quantum technologies is reshaping the methods by which computers communicate and process data, laying the foundation for the emergence of the quantum internet. For instance, recent experiments have demonstrated deterministic delivery of correlated photons between quantum memories at separate locations, utilizing telecommunication wavelengths to extend the transmission range [52], [165]. Satellite-based quantum communication has also emerged as a promising solution for long-distance quantum information transfer, extending the reach of experiments to thousands of kilometers or more [209]. These developments set the stage for a global quantum communication network that could revolutionize secure data transmission.

Unlike classical networks, which utilize bits represented by electrical signals or electromagnetic waves, the quantum internet employs qubits that can simultaneously exist in a superposition of 0 and 1 states. This feature, along with entanglement, enables certain types of computation that surpass the capabilities of classical computers, such as factorization and unstructured search [210].

Entanglement refers to qubits that are correlated in such a way that the state of one can instantly impact the state of another, regardless of distance. Furthermore, the fact that any measurement on qubits disrupts their current states provides what is known as “unconditional security.” These properties will allow the quantum internet to enhance security and computational capabilities. One of its most notable applications is QKD [49], a secure protocol that generates and distributes secret keys such that any eavesdropping attempt is detectable. This is particularly valuable for sensitive communications in sectors such as finance, healthcare, and national security [73].

Beyond security, the quantum internet opens up new fields of application, including DQC [211], quantum sensing [4], and private quantum clouds. DQC connects multiple quantum computers to share resources, significantly expanding computational power and supporting solutions for quantum data centers. Quantum sensing provides highly sensitive measurements that advance fields such as medical imaging, navigation, and environmental monitoring. Connecting quantum sensors in a network allows for the development of even more robust and sensitive systems. Quantum private clouds allow cloud computations to remain confidential, even from the cloud provider.

However, the framework for quantum internet architecture remains unsettled, outlining two dominant models [15], [21], [67]. The first model aligns with traditional networking practices, emphasizing point-to-point data transfer through direct qubit transmission over connected channels [18], [66]; we refer to this as a qubit-forwarding network. The second envisions an overlay network that distributes entanglement via intermediary nodes, allowing endpoints to use the final E2E entanglement for communication [116], [117]; this is referred to as an entanglement distribution network.

It is worth noting that, as QKD is a prominent application of the quantum internet, some literature uses the term "quantum network" to refer specifically to networks consisting of QKD channels, which are dedicated networks focused on distributing secret keys between endpoints. This chapter also considers these networks, referred to as QKD networks. In such networks, specialized relays are designed to forward secret keys generated through QKD [75], which enable classical encryption between remote endpoints.

Regardless of network type, there is a consensus that the quantum internet will operate as a hybrid quantum-classical network, integrating components of the existing classical internet infrastructure. It is intended to optimize performance and security, particularly during the NISQ era, where quantum systems remain sensitive to environmental factors and are limited by scale [10]. Nonetheless, the quantum internet remains in its developmental stages and faces numerous unique security challenges [27]. Key properties of quantum networks—such as superposition, entanglement, and the no-cloning theorem [46]—introduce novel vulnerabilities that traditional security measures cannot effectively address. For example, the no-cloning theorem, which prevents the exact copying of arbitrary unknown quantum states, introduces a potential vulnerability in data recovery and redundancy mechanisms, compared to classical systems that depend on data replication. Mitigating these vulnerabilities will require innovative security paradigms tailored specifically for quantum networks, capable of managing both the inherent uncertainties of quantum systems and their integration with classical infrastructure.

Additionally, while QKD effectively mitigates conventional Man-in-the-Middle (MiTM) attacks, it remains vulnerable to attack strategies that exploit quantum properties, such as Photon Number Splitting (PNS) [123], which compromises the security of weak coherent pulses in QKD systems, as well as time-shift attacks [212], which manipulates timing information to gain an advantage in

key extraction. DoS attacks further threaten it by disrupting quantum channels or overwhelming system resources [213]. Beyond QKD, the broader network infrastructure faces new security challenges. For example, current quantum network routing relies heavily on synchronized and centralized classical systems, making them susceptible to single-point-of-failure and synchronization-based attacks [214]. This centralized dependence undermines resilience and presents a significant vulnerability in the network architecture. Thus, developing robust security protocols and protective measures specifically for the quantum internet is crucial. Given the rapid advancements in quantum technologies and the emergence of new threats, these security measures must be continuously reviewed, revised, and adapted to ensure the quantum internet's long-term integrity and resilience.

To facilitate research in quantum internet security, this chapter provides an overview that not only compiles known vulnerabilities and mitigation strategies but also adds empirical value through simulations. It begins with a review of the core components of the quantum internet, followed by a layer-based structure that encapsulates the essential features of various quantum internet stacks [26]. We subsequently dissect vulnerabilities specific to each layer, offering initial detection, counteraction methods, and evaluation metrics governed by the key tenets of information security: CIA. It is worth noting that this chapter does not attempt to offer a definitive solution. Instead, it highlights its significance through analytical findings and introduces a security framework for future research. Overall, the main contributions of this chapter can be summarized as follows:

1. We provide a layer-by-layer breakdown of the specific security issues encountered in quantum networks.
2. We introduce various defensive strategies designed to address these issues across various layers.

3. We assess the severity of potential attacks and evaluate the anticipated effectiveness of mitigation methods through analyses and simulations.
4. We identify the vulnerabilities within different types of networks.

To better understand the content of this chapter, readers are encouraged to first review Section 1.4 and Subsection 1.5.3.

5.2 The Quantum Internet Protocol Stack

Like its classical counterpart, the quantum internet can be organized into distinct layers. We define a simplified layer-based structure based on the core features of various quantum internet stacks [26]: physical, link, network, and application. Rooted in recent advancements, these layers offer specific services and protocols, as detailed in Table 5.1. Note that each layer also incorporates components of classical networks.

Table 5.1. Protocol Stack of the Quantum Internet

Layers	**Components**	**Example Technologies**
Application Layer	Quantum cryptography	E2E QKD.
	Data transmission	Quantum teleportation.
	Large-scale computing	Distributed computing.
Network Layer	Quantum router	Routing protocols.
Link Layer	Quantum repeater scheme (direct-link protocol types)	Qubit forwarding. Entanglement swapping. QKD relay.
Physical Layer	Transmission medium (quantum channel and classical channel)	Optical fiber. Free space/Satellite.
	Quantum computer	Photon-based. Superconductor-based.

At the bottom, the physical layer forms the foundation of the quantum internet, consisting of storage, computation, and transmission devices. This layer generally includes quantum memories, computing units, and communication links. While there are different types of repeaters, quantum networks often employ systems based on photons, using optical fibers or open-air transmissions [15].

Above the physical layer, the link layer manages direct connections between neighboring nodes, i.e., handling qubit transmission, or generating direct-link entanglement between them. In this layer, qubit-forwarding repeaters manage the physical transfer of data and QEC between adjacent nodes. Entanglement-swapping repeaters create entanglement between non-adjacent nodes [52]. Additionally, this layer facilitates remote QKD processes in settings involving trusted relays. Overall, the link layer is crucial for maintaining the integrity and security of data as it moves from one node to another, ensuring that quantum information is accurately relayed, or that entanglement is preserved during transmission.

The network layer manages the routing and flow of quantum traffic throughout the network, which varies based on the types of repeaters used and may involve qubit transmissions or entanglement requests. Although applying traditional networking methods to qubit transmissions may appear straightforward, the process often requires many QEC qubits. This has made the entanglement-swapping routing method a key area of research focus. Nevertheless, the practical implementation of E2E entanglement routing is fraught with challenges, including limited coherence times and the complexities involved in unreliable operations. These issues have spurred further research into quantum routing protocols and resource management [116], [117], [170].

At the top of the stack, the application layer provides users with quantum applications and services, including quantum teleportation, sensing, and DQC. It is important to note that the placement of QKD within the network architecture can vary depending on the technologies used: in networks based on QKD relays, it is positioned at both the link and application layers, whereas in entanglement networks, it is considered part of the application layer.

5.3 Vulnerabilities and Countermeasures

In alignment with the quantum internet stack discussed above, we categorize quantum internet security risks into four main areas: physical, link, network, and application layer attacks. Each category contains attacks specific to its respective layer.

5.3.1 Physical Layer

Physical layer attacks often focus on quantum memories, communication resources, and other hardware components. These vulnerabilities typically stem from flaws or limitations in the hardware. In this section, we discuss attacks that target the physical layer.

Photon Number Splitting (PNS) Attacks

PNS attacks are frequently associated with the "prepare and measure" QKD protocols, such as BB84, which requires transmitting quantum states. These attacks, however, can target any photon-based qubit transmission, either wired or free space.

In PNS attacks, an eavesdropper capitalizes on the fact that a sender might occasionally emit multiple photons when using a weak coherent pulse source [123], [215]. This means some pulses may contain no photons, some a single photon, and some multiple photons. In the case of multiple photons, an eavesdropper could intercept some of them, measure them to gain information about

the transmitted state, and then pass the remaining photons to the receiver undetected, thus compromising confidentiality.

PNS attacks can be countered through various methods [216], [217], [125]. One direct approach is to use a true single-photon source rather than an attenuated laser. However, this technology is still evolving [218]. While a single-photon source has been successfully experimented with [219], it faces challenges due to relatively low efficiency. This inefficiency arises from the quantum nature of the emission process, where the probability of producing a photon at a desired time is inherently limited. This results in a lower rate of photon production compared to more conventional sources such as attenuated lasers.

An alternate strategy is decoy states [125], which use states (pulses) with varying intensities: signal states and one or more decoy states. The signal states are used for the key distribution, while the decoy states detect eavesdropping. The key idea is that the sender randomly varies the intensity of the laser pulses between signal and decoy states without informing the receiver about which pulses are which. The decoy states are typically sent at different, often lower, intensities than the signal states. The lower intensity of the decoy pulses means they are less likely to contain multi-photon instances, making them ideal for detecting eavesdropping. If an eavesdropper tries to intercept these pulses, legitimate users can detect the differences in the quantum statistics between the intercepted decoy and untouched signal pulses. By comparing the rates of quantum bit errors and other parameters between signal and decoy pulses, legitimate users can estimate the fraction of single-photon pulses and detect the presence of an eavesdropper. This analysis allows them to adjust their key generation processes to account for any potential information leakage and ensure the security of the final key.

Furthermore, employing MDI-QKD or entanglement-based QKD can also prevent such attacks. In MDI-QKD, Alice and Bob prepare an entangled pair and send one half to a third party, Charlie, who is not necessarily trusted. Charlie performs a joint quantum measurement on the incoming qubits from Alice and Bob, similar to entanglement swapping. He then publicly announces the outcome of his measurement, which does not disclose any specific information about the states sent by Alice and Bob. Rather, it only confirms whether the qubits from Alice and Bob are correlated in a specific way as per the BSM. Based on Charlie's announcement, Alice and Bob can identify which of their qubits were successfully entangled during the measurement and disregard any unentangled qubits. By leveraging the principles of quantum entanglement and nonlocality, and not depending on the reliability of the measurement devices, MDI-QKD and entanglement-based QKD are considered secure against PNS and other detector side-channel attacks.

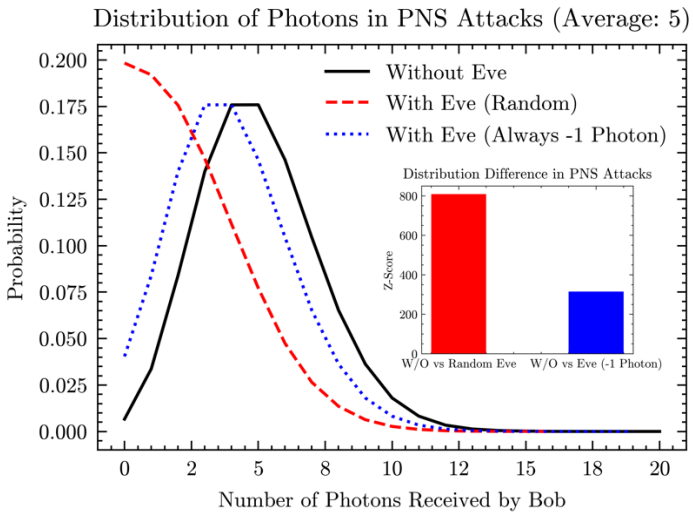


Figure 5.1. *Photon Distribution in the Simulation of PNS Attacks.*

Finally, to assess how a PNS eavesdropper's actions could influence their detection, we conducted a simulation analyzing the distribution of photons Bob receives under PNS attacks. In the simulation, Alice sends Bob photons distributed according to a Poisson distribution with an average value of 5, which mimics the random photon counts typical of weak coherent pulses. Eve,

the eavesdropper, is assumed to be able to intercept some photons and forward the remainder. We explored three different scenarios, depicted in Figure 5.1:

1. Without Eve: Bob receives all photons sent by Alice.
2. With Eve (Random): Eve randomly intercepts photons, and the rest goes to Bob.
3. With Eve (Always -1 Photon): Eve takes one photon, forwarding the rest to Bob.

Figure 5.1 illustrates how Eve's interference alters the photon distribution received by Bob. The Z-score graph within Figure 5.1 highlights the differences between the photon distributions, which indicates Eve's presence. A higher Z-score indicates a more substantial difference. Eve can hide better in the "Always -1 photon" setting compared to the "Random" setting (when only considering distribution differences), although the latter may provide more information. Bob, however, requires multiple transmissions to confirm this altered distribution.

Trojan-Horse Attacks

In the context of quantum networks, a Trojan-horse attack involves an attacker sending photons that appear harmless to the recipient but are embedded with harmful signals or configured in a manner that allows the attacker to extract information or disrupt the system when these photons are measured or processed by the recipient's equipment.

With unauthorized access to a quantum link, an attacker can initiate Trojan-horse attacks by sending a bright light pulse through the channel and examining the reflected pulses to gather information on the transmitted quantum states [220], [124], [221], [222]. This is possible because no optical component perfectly transmits or absorbs light, allowing some light to travel in the opposite direction of the main optical signal. This vulnerability enables such attacks, for example, to determine the measurement basis used in BB84 or acquiring information about transmitted

qubits. These breaches can compromise both the confidentiality and availability of communications.

Taking “prepare and measure” QKD as an example, to counter such attacks, Alice and Bob need to discard a substantial portion of their raw key through a process known as privacy amplification [123]. It involves distilling a shorter, highly secure key from the longer, potentially compromised raw key using specific mathematical functions (e.g., hash functions) to eliminate Eve’s partial knowledge. This may, however, reduce the secret key rate.

To reduce Eve’s photon gain without significantly lowering the key rate, Alice and Bob can employ phase randomization on the transmitted photons, introducing a layer of randomness and uncertainty to counter these attacks [223]. The phase of a quantum state is a critical characteristic that defines the state, along with the probability amplitude. While changing the phase does not affect the probabilities of measurements in the $|0\rangle$ and $|1\rangle$ basis, it influences the outcomes in other bases used to describe the state. By randomizing the phase, the quantum state undergoes unpredictable shifts. The altered phase would mask its true state if Eve intercepted the qubit. Alice and Bob can then synchronize their phase-randomized states over classical channels.

Eve must guess the measurement basis when she intercepts a reflected photon to extract information through these attacks. For clarity, let us define Eve’s information gain as the number of photons she successfully measures on the correct basis plus the partial information she obtains when measuring in an incorrect basis, denoted as G. That is, a correct guess provides a complete photon gain: $G + 1$. An incorrect guess yields a partial photon gain: $G + \gamma$ where $0 \leq \gamma < 1$.

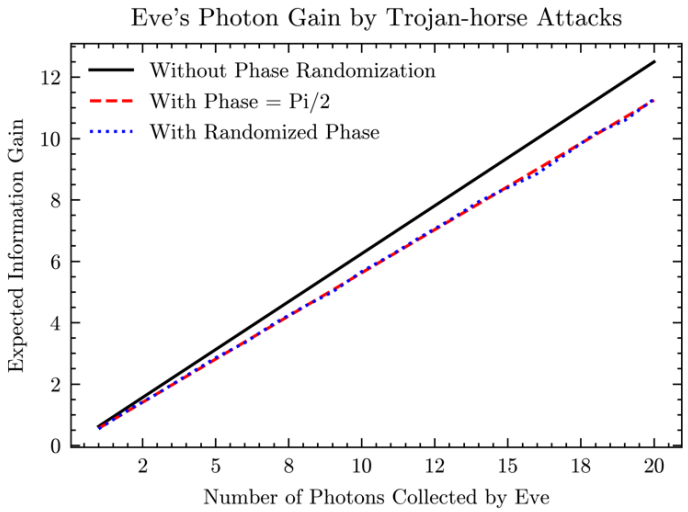


Figure 5.2. *Phase Randomization in the Trojan-Horse Simulation.*

In Figure 5.2, we plot Eve's photon gain as a function of the number of photons she collects during transmissions from Alice to Bob, comparing scenarios with and without phase randomization. In the simulation, Alice can send four possible states: $|0\rangle$, $|1\rangle$, $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and $|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. The states $|0\rangle$ and $|1\rangle$ are on a rectangular basis, while $|+\rangle$ and $|-\rangle$ are on a diagonal basis. As discussed, the photon gain can be calculated as follows:

$$G = \sum_{i=0}^{n} \gamma_i \quad (5.1)$$

where n is the number of photons collected by Eve and γ_i represents the incremental gain from measuring the ith photon, with $0 \leq \gamma_i \leq 1$. γ_i depends on Eve's probability of correctly guessing the measurement basis used by Alice and is given by:

$$\gamma_i = \begin{cases} 0.5, & b_i^E \neq b_i^A \\ 1, & b_i^E = b_i^A = 1 \\ \frac{1}{2}\cos^2\frac{\theta_i}{2} + \frac{1}{2}\sin^2\frac{\theta_i}{2}, & b_i^E = b_i^A = -1 \end{cases} \quad (5.2)$$

where b_i^E and b_i^A represent the measurement bases used by Eve, and Alice for the ith photon, respectively, with values from the set $\{1, -1\}$ where 1 is the rectangular basis, and -1 is the diagonal basis. θ_i is the phase shift Alice applies to the ith photon.

As mentioned, phase randomization does not influence the photon gain when using the rectangular basis but reduces the gain in the diagonal basis. Without phase randomization, Eve can precisely determine the state of a photon intercepted from the diagonal basis ($|+\rangle$ or $|-\rangle$). With phase randomization, each photon Alice sends on this basis receives a random phase shift, rendering states. $|+\rangle$ and $|-\rangle$ unpredictable (i.e., lowering Eve's gain). If Eve intercepts and measures such a photon, her chances of accurately identifying the state drop to $\frac{1}{2}\cos^2\frac{\theta_i}{2} + \frac{1}{2}\sin^2\frac{\theta_i}{2}$ because, depending on whether the state is $|+\rangle$ or $|-\rangle$, Eve's probability of being correct reduces to $\cos^2\frac{\theta_i}{2}$ or $\sin^2\frac{\theta_i}{2}$, respectively.

With that, this simulation compares scenarios without phase shifts, with a consistent $\pi/2$ phase shift, and with randomized phase shifts for each photon. Notably, if Eve intercepts a photon with a $\pi/2$ phase shift on a diagonal basis, her chance of a correct measurement is 50%. The results highlight the effectiveness of phase randomization in protecting against Trojan-horse attacks, with overlapping performance between randomized phase shifts and consistent $\pi/2$ phase shifts.

Other Attacks on Quantum Devices

In addition to the two well-known attacks discussed above, there are other side-channel strategies targeting the imperfection in current quantum devices, including interference or tampering with the physical components utilized in communications, such as light sources, detectors, or quantum storage, to risks. For instance, photon-based systems heavily depend on detectors to measure the

quantum properties of single photons. Yet, these detectors are susceptible to compromise through specifically tailored bright illumination [224], [225]. Experiments showed that Eve could manipulate the gated avalanche photodiodes in the systems using bright illumination, effectively blinding them and transforming them into classical, linear detectors [224]. This alteration allows Eve to control the detector outputs, potentially capturing the full secret key undetected. Countermeasures such as installing optical power meters at detector inputs and incorporating their readings into security proofs can help thwart such blinding attacks. Nonetheless, determining completely hack-proof configurations for detectors remains a challenging task.

Additionally, phase-remapping attacks involve an attacker manipulating the phase settings within a quantum communication channel. This manipulation changes the intended phase encoding of the qubits being transmitted between legitimate parties, potentially leading to incorrect decoding and creating vulnerabilities in communication. The attack was demonstrated through experiments using a standard commercial QKD system that relies on phase-encoded photons [226]. The phase modulators in the system were altered to show how an external attacker could affect the phase settings during the key distribution process. By adjusting the voltages applied to these modulators, the phases of the encoded qubits were changed without being detected by the standard security checks in QKD. This manipulation could cause the receiver to misinterpret the quantum states, leading to errors in the key extraction process.

Moreover, strategies against such attacks include using quantum tamper-evident seals [227]. The goal of quantum seals is to leverage quantum mechanics to detect and indicate any unauthorized access or tampering. However, quantum states are highly sensitive to environmental disturbances such as temperature, electromagnetic fields, and movements. While useful for detecting tampering, this sensitivity also makes the seal prone to false positives from non-malicious environmental

changes. Additionally, some setups incorporate multiple quantum systems as backup channels or systems to increase availability in case of system compromise or failure. If one system experiences a failure or a security breach, the network can automatically switch to an alternative system, thus ensuring continuous secure communication. For instance, the QKD network in Vienna utilizes various QKD systems, each based on different principles of quantum mechanics, such as measurement or entanglement [228].

5.3.2 Link Layer

Attacks at the link layer target the protocol that governs communications or direct-link entanglements. This underscores the necessity for robust link layer protocols to protect against unauthorized state interference during transmission. This protection includes secure state encoding and error detection to rectify transmission anomalies.

Entangling-probe Attacks

In an entangling-probe attack, the attacker replaces the initialized qubits in a user's quantum register with qubits entangled with those in the attacker's register. This manipulation allows the attacker to perform local operations on these qubits and potentially alter the output of the user's algorithm [229], [230]. For instance, after the replacement, the attacker establishes entanglement between the qubits in the user's register and the qubits they control. Suppose the user executes an algorithm like Shor's algorithm with their qubits. In that case, the attacker can interfere with the operations, leading the user to obtain a random result by manipulating the entangled qubit, thereby compromising the integrity and availability. Also, the attacker might remain passive and wait for the user to measure their qubit. Once the measurement occurs, the attacker, through their entangled qubit, can deduce the user's results, thus breaching confidentiality.

In another scenario, suppose a user, Bob, already maintains an entanglement with another user, Alice. As illustrated in Figure 5.3, if an external entity, Eve, gains access to Bob's quantum computer, she can entangle her qubit with the pair. This can be done by applying a CNOT gate to Bob's qubit and her own, as depicted in the circuit in Figure 5.3. Then, the qubits of Alice, Bob, and Eve become entangled in a tripartite state. Any interaction by one party affects the entire entangled state and influences the other parties involved. If Alice and Bob believe they are using a secure direct-link entanglement, Eve could interfere with or eavesdrop on their communications. For instance, if Alice and Bob use their entanglement to generate a secret key, Eve might also be able to access it.

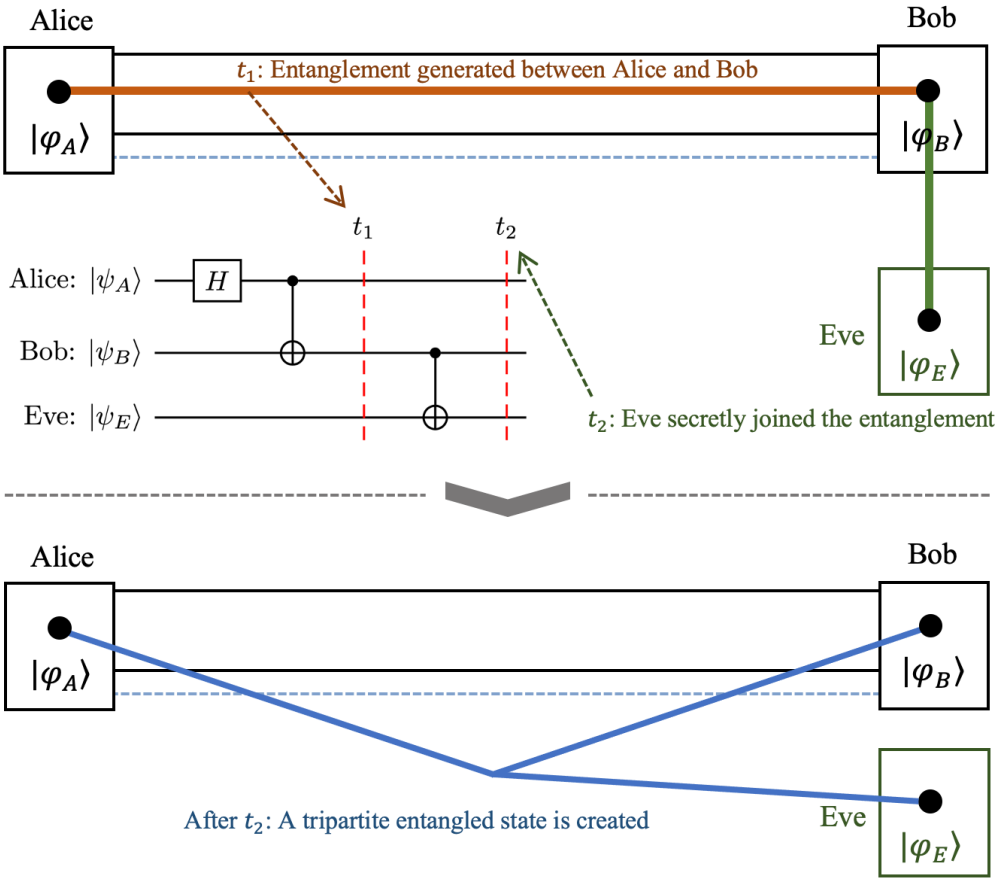


Figure 5.3. *An Infiltration with Entangling Probe.*

Additionally, there are potential methods for eavesdropping during the generation and delivery of entangled photon pairs. For instance, in typical direct-link entanglement generation scenarios, entangled photons are produced by a third party and then sent to Alice and Bob. If present, Eve could attack using either a polarizer or a decohering birefringent plate, both of which can be oriented and, in some cases, accompanied by additional wave plates to allow analysis in arbitrary elliptical polarization bases. Experiments have demonstrated that the presence of an eavesdropper can be consistently detected by checking for violations of Bell's inequalities [231].

Man-in-the-middle Attacks

In addition to entangling probe attacks, if an attacker gains access to a user's computer, they can launch MiTM attacks by impersonating the receiver or sender. For example, Eve can deceive Alice into believing she is communicating with Bob when she interacts with Eve. This deception can be accomplished by intercepting both classical and quantum channels. Eve can then manipulate the intercepted classical messages to align with her goals—copying, modifying them, or creating entirely new messages unrelated to Alice's original. She can also replicate the initial protocol used to re-encode the message intended for Bob in the next communication phase and send it to him.

In classical networks, authentication is a standard defense against MiTM attacks. However, authenticating in quantum systems presents challenges. Alternatively, Alice and Bob could use an interlock protocol [232], a method commonly discussed in classical networks for detecting the presence of Eve. In this protocol, Alice encrypts her message with a code that cannot be decoded instantly, and Bob does the same. Alice sends a portion of her encrypted message to Bob, and he sends a portion of his to Alice. Upon receiving Bob's partial message, Alice sends the remaining part of her encrypted message; similarly, upon receiving Alice's part, Bob sends the remainder of

his. The messages are then combined to complete the communication. This sequential exchange prevents Eve from altering any part of the message without having access to the entire message. If Eve attempts to modify and send a previous part to Bob, she must accurately predict the subsequent part that Bob expects. An incorrect prediction will result in a nonsensical reconstructed message, alerting Bob to potential tampering. While in classical networks, this method does not prevent Eve from merely listening and forwarding, in quantum networks, any attempt to read or copy the quantum state would collapse, revealing the interference [233].

Attacks on Error Correction

Qubit-forwarding transmission relies heavily on QEC, which introduces vulnerabilities. If adversaries know the specific QEC scheme, they can introduce noise to disrupt it. They might deliberately introduce errors just below the QEC code's tolerance threshold, potentially overloading its capacity and leading to logical errors. This critical point occurs when the rate of logical errors (errors affecting logical qubits) exceeds the rate of physical errors (errors affecting physical qubits).

One fundamental weakness is the susceptibility to specific types of errors [164]. For example, the 3-qubit bit-flip code, effective against single bit-flip errors, cannot handle multiple simultaneous errors. Also, it is ineffective against phase-flip errors that transform states, such as $|+\rangle$ to $|-\rangle$. Another vulnerability is that QEC's reliability hinges on maintaining an error rate below a break-even point. For the 3-qubit bit-flip code, this is $1/3$. Breaching this point results in ineffective error correction. Also, QEC relies on periodic checks using ancillary qubits. Any delay or failure in these periodic checks can go unnoticed, leading to an accumulation of errors. QEC is also

computationally intensive; an adversary could exploit this by forcing the system into near-constant error correction, essentially launching a DoS attack.

To address these vulnerabilities, a combination of QEC codes can be employed and alternated, complicating the ability of attackers to introduce noise [234]. Using redundant ancilla qubits, cross-referencing outcomes, and maintaining a secure environment are crucial strategies. Nonetheless, given the current hardware limitations associated with QEC in qubit transmissions, entanglement-swapping networks represent a more viable option in the near term.

5.3.3 Network Layer

At the network layer, attacks are aimed at disrupting operations and resources, including routing and entanglement generation. Routing strategies that depend on centralized control, for instance, are prone to single points of failure. This section details the vulnerabilities in quantum networking.

Untrusted Repeater Nodes

Most existing quantum routing schemes presume the use of trusted repeaters [116], [115], [66], [235]. However, this assumption introduces several vulnerabilities, particularly if an attacker gains access to a repeater or substitutes it with a counterfeit device. For instance, an attacker could deploy a fake quantum repeater or hijack an existing one to reroute traffic through a compromised system, akin to spoofing attacks in classical networks [236].

In qubit-forwarding networks, a breach of the quantum repeater would enable an attacker to intercept or manipulate all qubits passing through a compromised repeater. Once in control, the attacker could reroute network traffic, harvesting communication resources and injecting malicious payloads. Furthermore, diverting traffic through a malicious repeater could lead to

network inefficiencies or enable targeted denial of service attacks. Detection of such tampering in these networks is relatively straightforward, as any interference with the transmitted qubits would cause the qubits to collapse, which the receiver would notice. However, effectively preventing such attacks is challenging due to the scarcity of practical encryption methods for qubits [237]. Presently, efforts are focused more on transmitting a reliable and larger number of qubits. This vulnerability can be mitigated by utilizing entanglement-swapping repeaters, which do not transmit data but establish E2E entanglement, thus circumventing potential MiTM attacks once the entanglement is set up.

Nonetheless, in networks that use entanglement-swapping, an attacker could potentially disrupt the entanglement process within quantum repeaters. By interfering with the entanglement process, an attacker could introduce errors before swapping, thereby reducing the success rate of the swaps and consequently lowering the overall rate of E2E entanglement [238]. This could also allow an attacker to modify its operations, such as altering how routing is processed, which could sabotage or manipulate the creation of E2E entanglement. Moreover, a compromised repeater might secretly introduce a third qubit into a two-party E2E entanglement and forward this extra qubit to an eavesdropper, as discussed in Subsection 5.3.2. This could allow unauthorized parties to listen in on or interfere with the operations performed on the E2E entanglement.

Additionally, for the routing and swapping process to be complete, there needs to be an entanglement identifier and quantum repeater identifiers [239], [240], [241]. However, the inherent complexities of quantum systems, such as their sensitivity to disturbances and the presence of quantum noise, make it difficult to apply traditional control and identification methods. The lack of robust identification mechanisms for quantum systems exacerbates the problem [242], [243]. An attacker could impersonate a quantum repeater by forging identifications and undertaking

malicious activities such as selecting a longer or inaccessible path. They might send fraudulent addressing messages to quantum repeaters, linking a malicious address with that of a legitimate quantum repeater in the network. This could result in operations intended for the legitimate address being redirected and tampered with by the attacker.

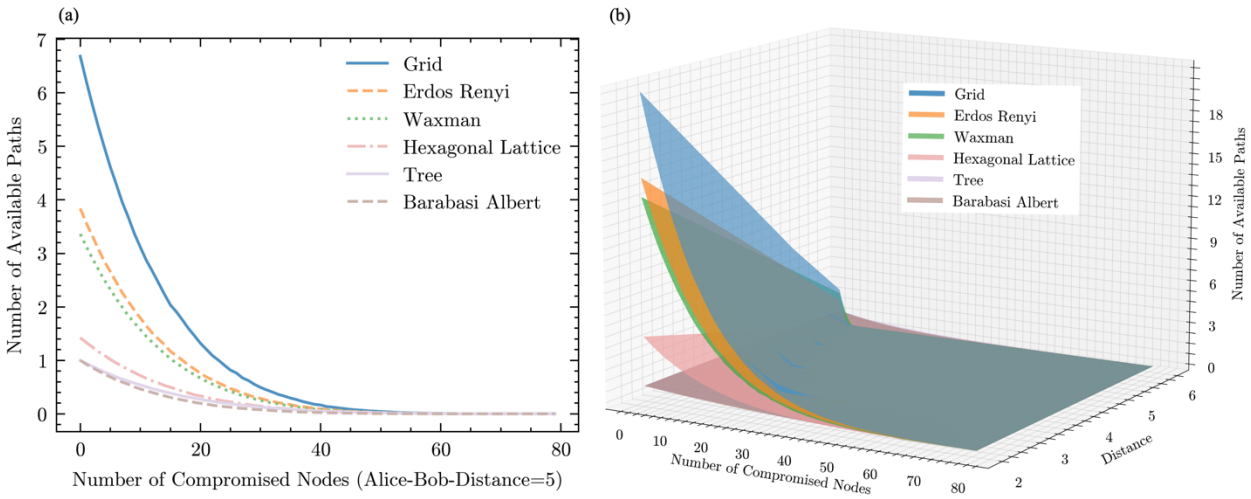


Figure 5.4. *100-node Simulations across Various Topologies (Topologies detailed in [244]).*

Parameters: Gird (10×10*), Erdős Rényi (probability for edge creation,* $p = 0.2$*), Waxman (model parameters,* $\alpha = 0.1$*,* $\beta = 0.4$*), Hexagonal Lattice (*6×7*), Tree (height* $h = 4$*, children of each node* $r = 3$*), Barabási Albert (number of neighbors,* $k = 3$*). (a) Decrease in path numbers as untrusted repeaters grow; (b) 3D plot by adding the distance dimension.*

Regardless, when a repeater is compromised and identified, a common solution is to bypass it, although this may introduce additional hops. We conducted simulations using various network topologies to assess the impact of bypassing untrusted repeaters on performance. We created networks with approximately 100 nodes in different configurations, including a 10×10 grid and a four-level balanced tree where each node has three children. We also generated random topologies using models such as Erdős-Rényi, Waxman, and Barabási-Albert [244]. In each network, we randomly designate certain nodes as compromised and exclude these nodes when finding viable paths. As expected and shown in Figure 5.4, the simulations indicated an

exponential decrease in available paths as the number of compromised repeaters increased. When the proportion of untrusted nodes surpasses a certain threshold—such as 60% of the network being compromised—all viable paths become unavailable, as shown in Figure 5.4(b). Moreover, some network topologies demonstrate more resilience to compromised nodes than others. For example, grid topologies, with their non-hierarchical structure and multiple connection routes, are inherently more robust against node failures or compromises than tree topologies.

The analysis assumes that compromised nodes have been identified; however, assessing a node's trustworthiness remains challenging. Thus, a set of Quality-of-Service (QoS) metrics for quantum networks is needed to identify abnormal network patterns that may indicate a compromised node [245], [235], [240]. For example, nodes with higher error rates and longer response times would be considered untrustworthy. These QoS metrics could include critical performance indicators like throughput, delay, and overall utilization of network resources, which may vary depending on the type of network. For instance, throughput could be measured by the number of qubits transmitted in qubit-forwarding networks. In contrast, in entanglement-swapping networks, it could be determined by the rate of E2E entanglement generation. Based on these criteria, network paths could prioritize nodes that meet specific reliability standards and put weights in pathfinding algorithms, similar to the common practices in classical networks.

Finally, it is worth noting that, in MDI-QKD setups, relays can be untrusted because the protocol addresses vulnerabilities related to detector side channels. The role of the relay is limited to announcing the results of the measurements rather than the measurements themselves, preventing potential eavesdroppers, including those with malicious intent operating the relay, from accessing sensitive information. However, these untrusted relays function merely as single repeater links connecting two adjacent nodes. Therefore, unlike trusted relays, MDI-QKD does not support QKD

over arbitrarily long distances [75]. For large-scale QKD networks, integrating both trusted and untrusted relays is expected to be essential.

Denial of Service

DoS attacks can target multiple layers within a quantum network. At the physical layer, such attacks may involve interference with the quantum communication infrastructure, such as intercepting or altering qubits. While the end parties can detect an attacker's presence, their only solution is to switch to a different channel, thereby disrupting the service of the original communication channel. At the network layer, DoS attacks could interfere with routing quantum information or flood the network with excessive connection requests, leading to degraded performance. In this subsection, we focus on DoS attacks in the network layer.

In connection-oriented quantum networks [170], [246], a queue of pending connection requests is managed by a centralized routing system. The system processes these requests either individually or in batches. An attacker can continually submit connection requests to exhaust network resources or inundate its processing capability. Although connectionless networks might mitigate this issue by allowing a request to avoid blocking other connection requests through a dedicated connection path, they are not entirely immune and still face similar threats when congestion occurs [246].

Filtering suspicious traffic that causes DoS at the physical layer is challenging in quantum networks. Although the quantum measurement property allows us to detect the presence of an eavesdropper, there is no straightforward method to eliminate them. Thus, the only strategy is to utilize backup channels for availability. However, DoS attacks can be countered at the network layer using traditional defense mechanisms, as these attacks generally originate from classical networks within hybrid quantum-classical systems. For instance, DoS attacks involving excessive

connection requests can be managed by scrutinizing these requests for abnormal patterns that suggest an attack. Once detected, the origins of these dubious requests can be isolated or redirected to a "sinkhole," where they are discarded. This approach mirrors the sinkhole technique used in classical networks [247].

Another widely used tactic involves rate limiting and implementing timeouts, which restrict the number of connection requests a single source can make within a specific timeframe. This method helps prevent both intentional and accidental DoS attacks originating from within the network by capping the number of incomplete or embryonic connections—a type of connection that has yet to complete the necessary classical communications between the source and destination, such as messages that communicate the results of direct-link entanglement generation [116], [115], or those detailing the instant network topology for routing purposes [117]. When the threshold for these embryonic connections is exceeded, further requests from the source are blocked until the network can verify or permit them again [248].

Integrating queuing or scheduling algorithms can improve the management of connection requests, mitigating congestion that might escalate to DoS. Adopting fair and efficient resource management strategies tailored to quantum QoS metrics is thus crucial [170]. For instance, schedulers could prioritize requests based on levels of suspicion to keep services running for legitimate users while minimizing the impact of an attack. This could involve assessing factors such as the frequency and type of requests a session generates and comparing these to the normal behavior patterns of legitimate users [249].

Similar to the solution to DoS at the physical layer, allocating backup resources for essential operations also helps the network sustain availability. By having multiple network paths and

routing servers, even if one route is compromised or overwhelmed by a DoS attack, other paths can handle the necessary connections. This redundancy helps balance the load and provides alternative routes for data to travel, reducing the potential for a single point of failure [250], but it comes at a higher cost.

Routing Disruption

Most quantum routing schemes, such as those used in qubit-forwarding and entanglement-swapping networks, rely on a classical centralized routing node to perform tasks such as pathfinding and managing quantum memories across lossy links [170], [17]. These routing nodes are vulnerable to attacks, particularly when they are overloaded with an excessive number of requests. Such attacks could fill the routing table and exhaust the nodes' processing and memory resources, increasing response times and potentially leading to DoS. The reconnection attempts of failed requests could exacerbate this situation. If routing nodes are hijacked or impersonated, attackers could manipulate the routing process to divert qubits through compromised nodes they control, potentially leading to eavesdropping or data alteration and thus compromising confidentiality and integrity. Attackers might also force data to take longer paths, increasing loss rates and hindering successful communication. They could even introduce "shadow nodes" that appear legitimate, drawing users to route data through them, similar to how rogue Wi-Fi hotspots attract devices with their strong signals [251].

For entanglement distribution networks, security concerns vary based on whether there is global or local link-state knowledge [116]. With global link-state knowledge, every node has a holistic view of the direct-link generation status before routing (i.e., a global view of the instant topology), which typically requires a centralized routing server and shares the vulnerabilities mentioned

above. With local link-state knowledge, nodes only have information about the direct-link entanglements with their adjacent nodes. Each node then indiscriminately swaps entanglements to establish an E2E entanglement between endpoints [252]. If an attacker hijacks a node, they can manipulate swaps to reduce the success rate, increasing delays and compromising availability. In distributed graph structures, such as those used in asynchronous routing schemes, certain nodes, like the root nodes of trees, may control the routing graph [117]. An attacker masquerading as a trusted root node could announce invalid routes, directing nearby user requests towards these non-functional routes and violating availability and integrity. This impact could be more severe if the attacker can hijack multiple routes across the network.

To combat these threats and ensure the integrity and availability of the network, it is crucial to implement redundant resources and robust detection mechanisms within classical networks, particularly when dealing with untrusted repeater nodes. For instance, employing a distributed routing system rather than relying on a centralized network node can avoid single point of failure [175], [253]. Additional protective measures like encrypting classical network traffic and implementing strong authentication and authorization are also vital. Additionally, attackers can target resource management algorithms that allocate quantum memories, attempting to monopolize these resources by mimicking criteria prioritizing connection requests. Reliable criteria and redundant resources discussed can help thwart these maneuvers.

Finally, an attacker could exploit vulnerabilities in time synchronization within quantum systems [160], altering the expected arrival times of entangled states and affecting entanglement-swapping outcomes. Many entanglement routing schemes depend heavily on event timing precision [116], [17]. Disrupting time synchronization could lead to discrepancies in the anticipated timing of entanglement generation, introducing errors and misinterpretations. One immediate solution is to

eliminate synchronization in routing and adopt asynchronous routing protocols, such as [117], [252], [253]. Additionally, utilizing multiple independent time sources can diminish reliance on a single synchronization point. Each node can choose the most reliable time signal available, or a consensus mechanism could be employed to establish the correct timing by aggregating inputs from various sources.

5.3.4 Application Layer

At the application layer, attacks mostly target vulnerabilities in quantum applications such as QKD, quantum algorithms, or DQC architectures. These attacks can take various forms, including tampering with the integrity of QKD, manipulating inputs or outputs of quantum algorithms, or disrupting the coordination of distributed quantum resources.

Quantum Algorithmic Attacks

Quantum applications often involve collaborative execution of algorithms or protocols, either by receiving information from one end or using E2E entanglement jointly [211], [254]. For instance, two distant nodes participating in a DQC setup might need to execute a joint computation. These computations often rely on entanglement to perform operations across the network. One example is executing a distributed version of Grover's search algorithm, where nodes share database segments and use entanglement to enhance search efficiency across the distributed system. In another scenario, two remote entities, such as a bank and its branch office, might employ QKD to establish a secure channel. Additionally, two laboratories might collaborate to teleport a quantum state from one to another, necessitating a pre-shared entangled pair. In this process, one lab prepares a quantum state and performs a BSM using its segment of the entangled pair. The measurement results are then communicated classically to the other lab, which applies a specific

transformation to their pair segment, effectively reconstructing the original quantum state. Such collaborative architectures introduce vulnerabilities similar to those encountered in classical distributed systems, including compromised nodes, challenges with consensus, and the risk of fault propagation [255].

Moreover, attacks on quantum applications often target protocols, algorithms, and architecture vulnerabilities. For instance, amplitude amplification in Grover's algorithm enhances the probability of desired states within a superposition, increasing their likelihood of detection and providing computational benefits [256]. However, while essential for the algorithm's effectiveness, this amplification can also inadvertently magnify any quantum states or operations errors. These errors might stem from imperfect quantum gates, environmental decoherence, or errors in initial state preparations, potentially concealing an attacker's operations. An attacker could exploit such vulnerabilities to sabotage the outcomes of a distributed Grover's algorithm [257]. The integration of QEC and fault-tolerant strategies is thus crucial [258]. Even though they increase the complexity of quantum operations, they are indispensable for realizing robust quantum computing systems.

Moreover, incorrectly encoding problem instances into quantum states can introduce significant vulnerabilities, particularly in the Variational Quantum Eigensolver (VQE) algorithm [106]. VQE is a hybrid quantum-classical algorithm that uses a parameterized quantum circuit to approximate the lowest eigenvalue of a Hamiltonian. This Hamiltonian represents the energy function of a system, and the algorithm aims to minimize this function to determine the system's ground state energy. It is particularly useful in quantum chemistry and material science. However, misrepresenting the Hamiltonian can lead the algorithm to optimize an incorrect model, resulting in flawed solutions [259]. This misrepresentation could be due to encoding errors or tampering,

making the system susceptible to manipulation. Such vulnerabilities highlight the need for rigorous validation and verification in quantum computing to ensure the integrity and accuracy of computational results.

Finally, quantum algorithms and protocols often depend on specific assumptions or approximations that simplify complex quantum challenges. If these assumptions prove incorrect or are misapplied, they can create vulnerabilities. For instance, QKD setups often assume that the equipment operates flawlessly and accurately represents quantum states without unintended defects. However, quantum devices are typically noisy, which can obscure the presence of an eavesdropper. When the end parties compare results, distinguishing between noise and actual eavesdropping can be challenging. Thus, given the fast-paced advancements in this field, it is essential to revisit applications and security strategies to address emerging vulnerabilities consistently.

Quantum Probes

In quantum networks that facilitate "prepare and measure" QKD, attackers can exploit vulnerabilities in the transmitted states to extract data [260]. There are three categories of such attacks: individual, collective, and coherent [261]. It is worth noting that while entanglement-based and MDI-QKD can potentially avoid such attacks, they are included here for completeness. This is because "prepare and measure" QKD remains an active area of research, and new attacks evolving from those discussed here continue to be developed [262], [263].

In individual attacks, Eve employs a quantum probe for each qubit sent between Alice and Bob, designing the probe to become entangled with the transmitted qubits. This entanglement allows her to extract information without being detected by Alice and Bob. The critical challenge for Eve

is to engineer her quantum probes and their measurements to minimize any disturbance to the qubits, as any significant disruption can increase the Quantum Bit Error Rate (QBER). An increased QBER would alert Alice and Bob to potential eavesdropping. After Alice and Bob disclose their measurement bases (which they use to decode the qubit), Eve measures her entangled probes to glean information about the transmission, potentially compromising or altering the quantum data. While this strategy is commonly associated with QKD, it can threaten the confidentiality of any quantum communication. The risk of these attacks can be mitigated by setting a threshold for acceptable interference or error rates and promptly responding if these thresholds are exceeded.

Collective attacks are more sophisticated than individual ones, involving Eve entangling her probes with blocks of qubits instead of single units. She then treats these entangled qubits as a collective system, enabling her to exploit correlations that may emerge from QEC processes and the exchange of quantum information between Alice and Bob, potentially leading to more substantial data extraction. Employing randomized sequences in quantum transmissions can complicate Eve's data extraction. When combined with advanced QEC codes, these measures can significantly reduce the information that Eve can extract [264].

Coherent attacks involve Eve treating all transmitted qubits as one unified system. A high-dimensional quantum probe entangles the entire quantum state being sent. She then carries out a joint measurement of this entangled state. It allows Eve to access all transmitted data simultaneously. To counteract these attacks, sending extra states or photons at various intensities makes it challenging for Eve to determine which states carry real information [217]. Frequently changing the patterns of quantum transmission can also disrupt coherent attacks, making it harder to treat the entire quantum transmission as a single, coherent system.

5.4 Discussion – Readiness of Quantum Attacks

The complexity of executing a quantum attack depends on various factors, such as technical requirements, hardware accessibility, and available computational resources. This section explores these aspects for the discussed attacks, emphasizing their feasibility and potential impact.

Most attacks on the physical layer, such as PNS or Trojan-Horse attacks, require access to specialized quantum devices, including precise photon detectors, quantum memories, and sources capable of generating specific quantum states. These tools are not widely available and often require significant expertise and controlled environments. While certain components for some attacks, such as bright light sources used in Trojan-Horse attacks, can be commercially obtained, executing them typically necessitates additional equipment—such as optical couplers, connectors, and precise detectors—that are not easily accessible. Additional attacks, such as manipulating gated avalanche photodiodes and phase-remapping, involve sophisticated control over quantum devices and their phase settings, requiring high precision and expertise. Moreover, attacks in this layer must be performed in real time, as quantum states are highly sensitive to environmental changes. Therefore, while feasible, these attacks are generally challenging for an outsider to execute.

At the link layer, direct quantum channels between nodes are potential targets for disruption. Such attacks are more resource-efficient, but similarly, there are simpler strategies to counter them. For example, injecting noise into the channel to compromise transmission does not necessitate additional equipment once access is gained for MitM attacks. Meanwhile, such attacks can be detected by employing an interlock protocol, as described in Subsection 5.3.2. More complex attacks, such as side-channel exploitation during link initialization and targeted attacks on error

correction protocols, leverage implementation flaws. These attacks are considerably more challenging, requiring high precision and advanced tools.

Attacks on the network layer, particularly those targeting hybrid quantum-classical systems, demand significant classical computing power. For instance, DoS attacks on quantum routing or synchronization rely on classical resources to flood the system with a high volume of requests or disruptions. In contrast, DoS attacks targeting QKD networks can be executed by gaining access to the QKD channel, which requires quantum apparatus rather than classical computing power. Alternatively, attackers may use classical computers to simulate quantum systems, gain insights, or prepare for interventions. However, the computational resources required for such simulations increase rapidly with the complexity of the quantum system.

Application-layer attacks, which focus on tasks including key exchange or computations, are often the most resource-efficient for adversaries. These attacks typically exploit vulnerabilities in protocols, algorithms, or system architectures and generally do not require additional hardware. For instance, targeting distributed quantum algorithms, such as the collaborative execution of Shor's or Grover's algorithms, involves exploiting protocol weaknesses, such as timing or consensus issues in distributed architectures. However, quantum probes targeted at QKD systems, which involve direct interaction with quantum systems to extract or disrupt information, require hardware and expertise, making them more challenging to execute.

Overall, executing attacks on the quantum internet requires varying levels of resources and expertise, depending on the target and method. While higher-layer vulnerabilities can be exploited with relatively accessible resources compared to lower-layer attacks, most still demand specialized

hardware and significant computing power. For ease of reference, we have compiled the readiness of the discussed attacks in Table 5.2.

Table 5.2. Readiness of Quantum Attacks

Layers	**Attacks**	**Requirements**	**Readiness**
Application Layer	Quantum Algorithmic Attacks	Access to quantum systems, access to classical controllers	High – Feasible with the algorithm or protocol knowledge and access to classical systems.
	Quantum Probes	Devices to interact with quantum systems and measure or store states	Low – Demands advanced equipment and expertise.
Network Layer	Untrusted Repeater Nodes	Control or compromise of network nodes	Low to Moderate – Exploits network trust, requiring access to intermediate nodes.
	DoS	Classical computing resources to flood requests	High – Relatively simple to execute using classical tools to overwhelm the network.
	Routing Disruption	Access to classical network systems, tools for intercepting classical data	High – Feasible with protocol knowledge and access to classical control systems.
Link Layer	Entangling-probe	Access to quantum channels, entanglement sources	Low – Requires access to endpoints and advanced hardware to generate entanglement.
	MitM	Access to quantum channels, noise injection devices	Moderate – Feasible with access but challenging to execute without detection.
	Attacks on QEC	Access to quantum channels, noise injection devices	Moderate – Requires access to endpoints.
Physical Layer	PNS	Quantum non-demolition devices, quantum memory, precise photon detectors	Low - Requires advanced quantum-specific hardware and precise manipulation of photon states.
	Trojan-Horse	Bright light sources, optical couplers, precise photon detectors	Moderate - Some components are readily available.
	Phase-Remapping	Phase modulators, polarization controllers	Low - Demands precise control over phase settings.

5.5 Summary

The quantum internet offers enhanced communication and DQC opportunities, yet it also poses unique security challenges inherent to quantum mechanics. These challenges span various layers, including physical, link, network, and application. A thorough, multi-layered security investigation that addresses concerns from various perspectives is essential. This chapter conducts a layer-wise security analysis to stimulate further research into quantum internet security, paving the way for a secure and advanced future internet.

Chapter 6: Quantum Blockchains

Blockchain technology has been prominent recently due to its applications in cryptocurrency. Numerous decentralized blockchain applications have been possible due to blockchains' nature of distributed, secured, and peer-to-peer storage. One of its technical pillars is using public-key cryptography and hash functions, which promise a secure, pseudo-anonymous, distributed storage with non-repudiation. This security is believed to be difficult to break with classical computational powers. However, recent advances in quantum computing have raised the possibility of breaking these algorithms with quantum computers, thus, threatening the blockchains' security. Quantum-resistant blockchains are being proposed as alternatives to resolve this issue. Some propose to replace traditional cryptography with post-quantum cryptography—others base their approaches on quantum computer networks or quantum internets. Nonetheless, a new security infrastructure (e.g., access control/authentication) must be established before any of these could happen.

This chapter provides a theoretical analysis of the quantum blockchain technologies that could be used for decentralized identity authentication. We put together a conceptual design for a Quantum Blockchain Identity Framework (QBIF) with technical evidence. We investigate its essential components and feasibility, effectiveness, and limitations. Even though it currently has various limitations and challenges, we believe a decentralized perspective of quantum applications is noteworthy and likely.

6.1 Introduction

Blockchain technology, as a peer-to-peer technique that keeps track of sequential data, and its numerous applications, e.g., cryptocurrencies, has attracted massive attention in recent years. Besides cryptocurrencies, it provides an innovative way to realize decentralized applications,

which has led to many interesting research topics, such as decentralized consensus mechanisms, trustless security, cryptographical access controls, and non-repudiation [265]. Blockchain applications span various sectors, including finance, healthcare, the Internet of Things, cybersecurity, and many others [266], [267].

Public key cryptography and hash functions are one of the blockchains' main pillars to realize secure access control and distributed data. Public key cryptography schemes generally use two keys for encryption/decryption, user authentication, and digital signatures for secure data transmissions. These schemes require public and private key pairs that can be self-generated or obtained from a Public Key Infrastructure (PKI) consisting of a hierarchy of certificate authorities. In blockchains, they are used to authenticate network nodes. For example, Bitcoin [268] and most other existing blockchain platforms use the public-key method for Elliptic Curve Digital Signature Algorithm (ECDSA) for user authentication and access control. Also, hash functions are used to securely "chain" the data stored in blocks by referencing each block to its previous/parent block using the previous block's hash. For example, SHA-256 is used as a hash function to chain the data blocks in Bitcoin.

Public key cryptography and hash functions are prevalently used and are secure enough against factorization attacks and brute force attacks using classical computers. This is due to the difficulty of solving large-number factorization and reversing one-way functions. However, quantum computers have provided polynomial-time solutions to solve factorization (e.g., Shor's algorithm) [269] and efficient ways to reverse one-way hash functions in $O(\sqrt{n})$ (e.g., Grover's algorithm) [256]. In addition, such algorithms can also be used to find hash collisions efficiently, which could result in data tampering in blockchains [270]. Recent studies have shown the significance of tackling potential quantum threats to classical blockchains [271], [272], [273].

To address the potential quantum threats, new classical cryptography algorithms, called post-quantum cryptography, are being developed [274]. They use cryptographic primitives other than factorization, such as lattice, for public key schemes to avoid potential threats based on Shor's algorithm. Blockchains use public key schemes to secure their user accounts (public keys to generate addresses and pass codes based on private keys). If such public key schemes are endangered (e.g., pass codes can be found), user assets (e.g., cryptocurrencies) will no longer be secure. Post-quantum blockchains that use post-quantum public key cryptography have been developed and are claimed to be immune to quantum attacks [275]. They replace blockchains' current public key cryptography with post-quantum counterparts. On the other hand, the vulnerability of hash functions is not as impactful since we can increase the hash length to resist quantum attacks (e.g., a hash collision that maps two inputs to the same hash value). However, increasing the length sacrifices performance.

Moreover, quantum blockchains that are solely (or in a hybrid manner) based on quantum computer networks (or the so-called quantum Internet) are also being explored [276], [277], [278]. The classical data chain has been re-designed in a quantum manner by correlating quantum states over entanglements. "Unconditional security" of quantum decentralization is claimed for such quantum blockchains [279], [280].

However, quantum blockchains are still in their early phase and thus still lack sufficient research and implementation on the infrastructures. Quantum blockchain infrastructures such as access control and user authentication must be established before any quantum decentralization. Hence, in this chapter, we theoretically explore the essential components of a decentralized quantum application based on quantum blockchains.

The contributions of this chapter are listed below:

1. We provide a theoretical analysis of quantum blockchains and their use for a decentralized identity authentication application.
2. We accommodate the classical blockchain identity management with quantum advantages against the quantum threats (to classical cryptography).
3. We put together a conceptual design for QBIF and give a review of the related technologies.
4. We investigate the feasibility, effectiveness, and limitations of QBIF's essential components.

Even though it comes with various limitations and challenges, we believe a decentralized perspective of quantum applications is noteworthy and likely. Identity management frameworks promise to maintain users' identity information securely, and decentralized identity removes the concerns of centralization that violates user privacy [281]. Layering it to the quantum domain gets rid of the quantum threats. We believe that making blockchain quantum native and rebuilding everything from there would be an ideal way to implement quantum blockchains and their applications, but the current quantum technologies do not allow us to do that. This chapter analyzes the classical-to-quantum transition from a more realistic perspective by adopting existing technologies.

6.2 Background and Related Work

In this section, we introduce blockchains, quantum blockchains, and blockchain-based identity applications.

6.2.1 Blockchains and Quantum Blockchains

A blockchain is a shared and immutable ledger that maintains data in the form of transactions [268]. Transactions are linked by a Merkle tree and stored in blocks chained by hash functions. A blockchain is composed of a peer-to-peer distributed network where each node keeps a copy of a growing chain of blocks, and the network uses a consensus protocol to add new blocks and agree on the sequence of these blocks.

In general, blockchain users' (or clients') broadcasting of transactions starts a blockchain process. A new block with the latest transactions is made according to a particular consensus protocol (e.g., Proof-of-Work, Proof-of-Stake) [265]. Several network nodes may compete to produce the new block for incentives in these consensus protocols. For example, in Bitcoin, the creator (called "miner") of a block that is eventually accepted as a permanent part of the chain wins the competition and is rewarded. Each new block points to a previously created block that the creator of the new block accepts. Thus, a block followed by the longest chain in Bitcoin is considered final. Other blockchains use other similar methods for consensus.

As discussed, every block has a hash value and points to its previous hash value. If one of the blocks is tampered with, all its following blocks become invalid. Blockchain nodes discard this tampered copy of the chain and keep working on the chain that the majority keeps. By adding extra data to block headers or transactions, decentralized storage can be achieved. Together with the blockchain event triggers, smart contracts [282] and decentralized applications have been widely developed.

User access control in blockchains relies on public-key cryptography. Blockchain users keep their private keys as passwords to their accounts and use their public keys as account numbers.

Transactions usually include their generators' (e.g., payers in cryptocurrency) signatures which are done using their private keys. Network nodes that produce blocks verify all the new transactions by validating the signature with the generators' public keys and discarding invalid ones.

Commonly used public-key cryptography methods (e.g., ECDSA, RSA) rely on the difficulty of factoring large numbers. It is easy to compute the product of two large numbers, but it is challenging to factorize a large number. On the other hand, the reliability of hash functions mainly depends on the difficulty of reversing a one-way function, such as SHA-256 and MD5. It is easy to compute the hash with inputs but hard to invert given the hash. Every different input generates a unique hash.

However, public-key cryptography and hash functions are exposed due to the rapid development of quantum computers. Shor's algorithm can factorize large numbers in a polynomial time, endangering public key cryptography. Grover's algorithm and its derivatives provide methods with a high probability of finding the input to one-way functions by brute-force searches ($O(\sqrt{n})$, i.e., quadratic speedups).

Post-quantum blockchains and quantum blockchains have been proposed to solve these threats. Post-quantum blockchains replace blockchains' current cryptography with classical post-quantum cryptography. For example, post-quantum cryptographies have been proposed to replace ECDSA in blockchains (i.e., classical blockchains) with quantum attack-resistant methods [279], [283], [284]. Quantum-secured blockchains (i.e., hybrid blockchains) have been designed on top of QKD networks [285], [286]. Quantum blockchains (i.e., fully quantum blockchains) rebuild the classical blockchains in the domains of quantum computing. The notion of a quantum blockchain is

generally inspired by the non-separability of quantum entanglement. The sensitivity of data tampering (i.e., tampering transaction history in a block) is much stronger in quantum blockchains than in classical blockchains. Single tampering of a block would cause the denial of the whole chain, which indicates a higher level of security but a potential denial of service on a blockchain node. Note that, by tampering, we mean tampering with the blockchain historical data, not tampering with transmission information in general, e.g., the man-in-the-middle attack. However, man-in-the-middle attacks can happen in every aspect of a classical network activity (at any communication link). For example, an attack can be implemented during a key-generation communication or any kind of communication that may leak a secret key. If a key is leaked or intercepted, the blockchain account associated with that key will be compromised. In the quantum realm, such attacks will not be applicable.

Furthermore, a hard fork in a blockchain is another infamous issue. It sometimes happens due to disagreements in the blockchain community (i.e., intentional hard fork). For example, some want to upgrade the current blockchain (e.g., to increase the block size limit), but others disagree. Then, a new version of the current blockchain appears and is kept by users who believe in it (i.e., it becomes a hard fork). On the other hand, a hard fork happening due to technical problems (i.e., accidental hard fork) is rare because it will eventually be resolved by the consensus mechanism (i.e., the majority choosing the longer chain after some subsequent blocks being produced). Hence, a typical consensus algorithm can address it regardless of whether it is a quantum or classical blockchain.

6.2.2 Blockchain-based Identity Applications

Decentralized identity management has been one of the most prevalent applications of blockchains [287]. In the current internet, we have imperfect systems for digital identities that are kept in

centralized storage. Centralized institutions or companies must be trusted to maintain the authority of identity authentication. Centralized identity violates user privacy and might suffer from single-point-of-failure along with many other centralization challenges. One data breach would cause the leak of all confidential information. Also, user information could be manipulated by the centralized entity for its own interests.

Blockchain-based identity management has been proposed to tackle the above issues [281]. Every network node has the same source of truth to follow in a blockchain-based application. It replaces the centralized entity and provides flexible identity authentication. Digital identities are created as digital watermarks that are assigned to transactions. With a smart contract, such identities could be used as authentication methods for different purposes (e.g., financial and online activities).

There are usually three roles in blockchain-based identity management: owners, issuers, and verifiers. Issuers are trusted parties (e.g., local governments and companies). They can issue credentials (e.g., personal IDs and user profiles in an application) for users (i.e., owners). Owners keep the credentials to themselves and only provide them to third-party entities (i.e., verifiers) to prove their statements (e.g., age, ownership). Actual user credentials are not stored on blockchains. Only user statements' attestations (i.e., proofs) are kept on blockchains. For example, the proof of a user's date of birth may be stored on blockchains instead of the actual date of birth. When the user provides the date of birth and the proof, verifiers validate the issuer's signature and decide the genuineness of the user-provided data.

Blockchain-based identity applications have been prevalent for years. For example, Bitnation [288], a blockchain-based borderless voluntary nation project, has been developed to reclaim the sovereignty of identity information and governance services and has been working towards the

vision of world citizenship by registering identity information on the global blockchain. ConsenSys [289] has provided decentralized software services and applications. It has created a blockchain-based identity system that maintains decentralized identifiers for people, organizations, and objects on the blockchain platform Ethereum. Start-ups and big companies also have proposed many blockchain-based identity applications, such as Onename [290], ShoCard [291], and Ana. They used public blockchains to keep track of users' identity attestations, which were claimed to be invulnerable and easy to use. Moreover, to leverage biometric authentications for blockchain approaches, research has accentuated mobile hardware performance, such as energy consumption and computational delay [292].

Even though blockchain-based identity approaches still rely on the trust we hold in the identity issuers, which may escalate into a circular trust problem when some of the issuers are compromised, we believe decentralization does, to a certain extent, resolve the centralized ownership of personal identities. For example, companies and organizations will no longer own their users' or customers' personal information. They only know if their users satisfy a particular requirement. One can choose only to reveal enough amount of their identity attributes for verification. In other words, decentralization can maximize the need to preserve the users' or customers' privacy.

6.2.3 Related Work

Quantum decentralization has been discussed for years since the vision of quantum networks arose. QKD has been developed using hybrid quantum-classical networks, based on which decentralized quantum approaches (e.g., quantum blockchains) have been developed. For example, a data transmission protocol based on QKD, specifically BB84, and distributed ledgers has been proposed to maximize the security of key transmission [293]. Quantum computing naturally avoids

some classical attacks without extra effort. For instance, in [294], a decentralized quantum cash system, qBitcoin, has been proposed based on quantum teleportation, which naturally prevents double-spend attacks. In [295], the authors propose a quantum blockchain scheme with the concept of a quantum coin based on quantum entanglement, no-cloning theorem, and a delegated proof of stake consensus protocol. In [253], a decentralized base-graph routing protocol has been defined based on quantum repeater networks. It allows finding the shortest paths in quantum networks using only the local knowledge of the network nodes. Other methods of adapting quantum technologies to different blockchain components and their advantages in a quantum setting have also been explored [296].

There are various quantum approaches to achieving genuine authentication, such as quantum fingerprinting [297] and single-photon authentication [298]. An authenticator can determine whether a quantum identity matches a real one in multiple ways. For example, the quantum fingerprint is analogous to the classical digital fingerprint that associates a long string (an original string) with a much shorter string (its fingerprint), such that any two different long strings can be distinguished by only comparing their fingerprints.

However, there is no existing research on authentication applications based on quantum blockchain technology. As quantum computers are threatening the classical approaches, these applications need a way to adapt to the new challenges in the quantum era. Access control and authentication issues should be solved before decentralized quantum applications can be achieved. This chapter explores the feasibility of migrating the classical blockchain-based identity framework to quantum blockchains and proposes a conceptual design.

6.3 Technical Components

In this section, we introduce and review the technologies that QBIF is based on.

6.3.1 Quantum PKI

A user in a PKI has two types of keys: a public key that can be shared with anyone and a private key that is only kept by the user. The public key and private key are always in a pair. A public key can be used to encrypt a message, and only the corresponding private key can be used to decrypt it. For example, when Alice wants to send a secret message to Bob, Alice can use Bob's public key to encrypt the message and send it to him. Then, only Bob, who has the private key, can decrypt the message and see the plain text. Moreover, a private key can also be used to sign a message for non-repudiation. For example, when Alice wants to prove that she is the sender of a message, she uses her private key to sign the message before publishing it. Then, anyone with Alice's public key can verify the source of the message by matching it.

PKI is widely used to authenticate network activities (e.g., SSH, HTTPS, blockchains, digital identities). For identity management, it is used to generate a unique digital identity for a user by binding the user information (e.g., username, ID) with a public key. The binding is usually established by a Certificate Authority (CA) through the registration or sign-up process. Moreover, by signing with the user's private key, a user can prove the ownership of a message. By signing with a CA's private key, a CA can issue a statement (e.g., an attestation about an identity).

Technically, PKI is based on one-way functions, typically trapdoor functions. A trapdoor function, say, $f(x)$, is a function that, when given $f(x)$ and the secret s, it is easy to calculate the variable x. Without the secret s, it would be extremely difficult or impossible to get x. RSA is one well-known example that is based on trapdoor functions. Since the classical trapdoor functions are

endangered by the development of quantum computers and algorithms (e.g., Shor's algorithm and Grover's algorithm), quantum-enhanced trapdoor functions have been proposed, which led to the development of quantum PKIs. They use different types of keys by design (e.g., quantum keys in the form of quantum states or classical keys that consist of classical sequences of bits).

Either with quantum keys or classical keys, the encryption and signature schemes for quantum states enable quantum PKIs. For example, in [299], quantum keys based on a quantum one-time pad are used to encrypt quantum states. Moreover, a symmetric-key encryption scheme (where the keys for encryption and decryption are the same) for quantum states (called "private quantum channels") has been developed by using a pre-shared single use secret key (also referred to as one-time pad) [300]. It constructs a private communication of quantum states on an insecure one-way quantum channel (using a $2n$-bit classical key to securely transmit n qubits) [300]. Such private channels enable asymmetric-key encryption and, thus, quantum PKIs. For instance, in [285], a one-way quantum identity authentication has been developed to encrypt quantum states with classical keys. In [301], public-key encryption and authentication have been developed using quantum public and classical private keys.

Here, we briefly review the quantum PKI schemes presented in [301], [302] for quantum-state encryption, signature, and authentication. The notations introduced here will be re-used for our conceptual design of QBIF presented in Section 6.4. The schemes are based on trapdoor One-Way Quantum Transformations (OWQT) [301], [302]. Trapdoor OWQT is a quantum variant of a classical trapdoor function. The following paragraphs explain how the public and private keys are made based on a trapdoor OWQT and how they are used for quantum encryption, signature, and authentication.

Public-key encryption: The definition of the trapdoor OWQT U_{OWQT} [302] can be represented as (see Figure 6.1):

$$U_{OWQT}(|\rho\rangle\,|\psi\rangle\,|0\rangle\,|0\rangle) = |\rho\rangle\,|0\rangle\,|g(\psi,\rho)\rangle\,|f(\psi,\rho)\rangle \qquad (6.1)$$

where $f(\psi,\rho)$ is a classical trapdoor one-way function with a random parameter ρ. $g(\psi,\rho)$ is another classical function. $|0\rangle$ is an initial qubit. Particularly, $f(\psi,\rho)$, $g(\psi,\rho)$, and ρ constitute a further trapdoor function as a whole, say, $h(\psi,\rho)$ [302]. $h(\psi,\rho)$ is then used to implement the quantum trapdoor OWQT U_{OWQT}. $h(\psi,\rho)$ should satisfy:

1. Easy-to-operate: given a random parameter ρ as the trapdoor of $f(\psi,\rho)$ and $g(\psi,\rho)$, both $f(\psi,\rho)$ and $g(\psi,\rho)$ can be computed efficiently.
2. Hard-to-invert: given the values of $f(\psi,\rho)$ and $g(\psi,\rho)$, both ψ and ρ cannot be efficiently computed.
3. Easy-to-invert with the trapdoor: there exists a trapdoor (secret information) s for the trapdoor function (the trapdoor OWQT U_{OWQT}). Given the trapdoor s, the message ψ can be efficiently computed from $f(\psi,\rho)$ and $g(\psi,\rho)$, and ρ can be efficiently computed from ψ, $f(\psi,\rho)$ and $g(\psi,\rho)$.

In other words, with secret s, it is easy to compute ψ from (6.1) but extremely difficult otherwise (i.e., one-way function).

Suppose we have found such function $h(\psi,\rho)$ and we now want to use it to implement the trapdoor OWQT U_{OWQT} as a quantum circuit. As shown in Figure 6.1, we can implement it by translating $f(\psi,\rho)$ and $g(\psi,\rho)$ to three oracles U_f,U_g, and U_{fg} [302]. By cascading the three oracles, we get the quantum circuit for the trapdoor OWQT U_{OWQT}.

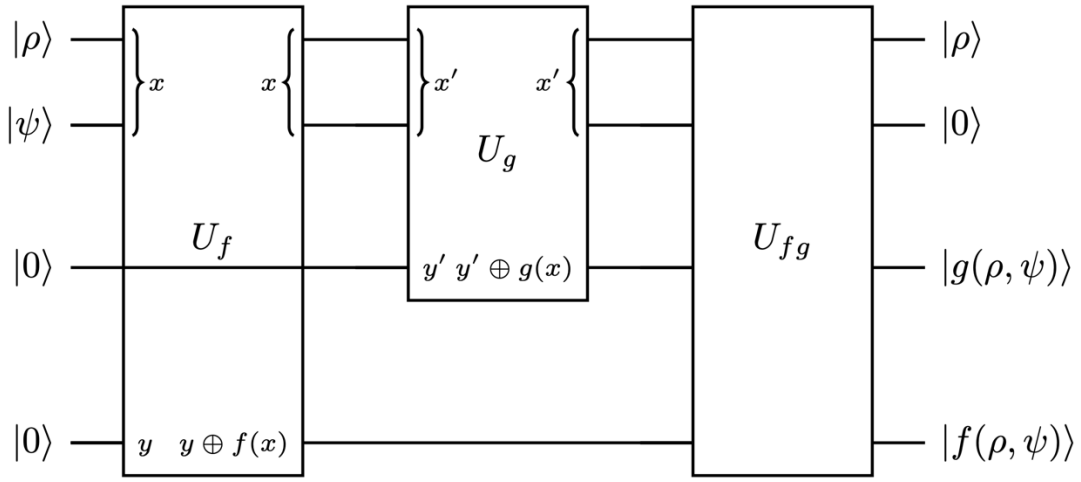


Figure 6.1. *The General Quantum Circuit for a Trapdoor OWQT.*

With the trapdoor function OWQT U_{OWQT}, the functions f and g are public. They can be prepared by the receiver so that only the receiver has the trapdoor s. Thus, the functions of f and g can be considered as the receiver's public key (or the tools to create public keys) and the trapdoor s as the receiver's private key. With these key generation processes, encryption and decryption of a quantum state $|\psi\rangle$ can be done as follow:

1. Bob finds a trapdoor function h and announces its components f and g as a public key. He keeps the trapdoor s to himself.
2. Alice wants to send an encrypted quantum message $|\psi\rangle$ that only Bob can read. She generates a U_{OWQT} according to Bob's public key.
3. Alice adds the parameter ρ and two initial qubits to $|\psi\rangle$ and gets $|\rho\rangle|\psi\rangle|0\rangle|0\rangle$.
4. Alice encrypts$|\rho\rangle|\psi\rangle|0\rangle|0\rangle$ by the U_{OWQT} (6.1) and obtains the ciphertext: $|\rho\rangle|0\rangle|g(\psi,\rho)\rangle\ |f(\psi,\rho)\rangle$.
5. Alice sends the ciphertext to Bob.
6. Bob decrypts the ciphertext with s.

There are two cases of decryption [302]. Here, we directly show the decryption transformation as

$$D_{fg}(|s\rangle|0\rangle|0\rangle\,|g(\psi,\rho)\rangle\,|f(\psi,\rho)\rangle) = |s\rangle|\rho\rangle|\psi\rangle|0\rangle|0\rangle \qquad (6.2)$$

Signature: Even though we have the public key and private key pair, the quantum digital signature is unlike the classical digital signature. Classically, we can sign a message and send it out. Anyone with our public key can verify the message's source without our intervention. However, the quantum signature process introduced here is an interactive digital signature protocol.

Suppose Alice has a trapdoor function h, consisting of f and g (6.1). Here, we represent the variables as bit strings: $h(\{0,1\}^{k+l}) = \{0,1\}^{k'+l'}$, where $\{0,1\}^{k+l}$ is a bit string of length $k+l$ with $\rho_1 \in \{0,1\}^k$ (k bits) and $\rho_2 \in \{0,1\}^l$ (l bits). Similarly, $\{0,1\}^{k'+l'}$ consists of $\rho_{\mathrm{r}} \in \{0,1\}^{k\prime}$ and $\rho_{\mathrm{s}} \in \{0,1\}^{l\prime}$. That is:

$$h(\{\rho_1,\rho_2\}) = \{\rho_r,\rho_s\} \qquad (6.3)$$

where $\{\mathrm{a},\mathrm{b}\}$ denotes the concatenation of two bit-strings, a and b. As discussed, h can be considered a public key. In this case, Alice has the private key (the trapdoor, say, s_h). The process to sign $|\psi\rangle$ and verify the signature is as follows:

1. Bob generates $\rho_{\mathrm{r}} \in \{0,1\}^{k\prime}$ randomly.
2. Alice generates $\rho_{\mathrm{s}} \in \{0,1\}^{l\prime}$ randomly and receives ρ_{r} from Bob. Now Alice has ρ_{r}, and ρ_{s} and the trapdoor s_h. These can be used to compute the inverse of h: $h^{-1}(\rho_r,\rho_s) = (\rho_1,\rho_2)$. Alice then signs the quantum message $|\psi\rangle$ with ρ_1 and function h (i.e., prepared as U_h) as in (6.4) and sends it.

$$U_h|\psi\rangle|0^{\otimes(k'+l')}\rangle = |\psi\rangle\,|h(\{\psi,\rho_1\})\rangle \qquad (6.4)$$

where $|0^{\otimes(k'+l')}\rangle$ denotes a state prepared with $k'+l'$ number of $|0\rangle$. For example, $|0^{\otimes 3}\rangle$ is short for $|000\rangle$.

3. Bob notifies Alice of the receipt of the quantum message.
4. Alice announces ρ_1 and ρ_2.
5. To verify the signature, Bob computes $h(\rho_1, \rho_2)$ (h is public) and checks if the first k' bits (of $\{0,1\}^{k'+l'}$) are ρ_r. If yes, it proves Alice has the trapdoor s_h of h. Bob then performs the transformation as in (6.5) to obtain the message (h, ψ and ρ_1 are known to Bob by this step).

$$|\psi\rangle\,|h(\psi,\,\rho_1)\rangle \rightarrow |\psi\rangle\,|0^{\otimes(k'+l')}\rangle. \qquad (6.5)$$

Finally, Bob accepts the signature only if the last $k'+l'$ qubits are all in state $|0\rangle$ (h belongs to Alice). Any signature that uses h to generate $|h(\psi,\rho_1)\rangle$ without the interactive process would be invalid. Note that this signature protocol does not support multiple verifications. The signed quantum state $|\psi\rangle\,|h(\psi,\rho_1)\rangle$ is invalid after the quantum message $|\psi\rangle$ is extracted. Therefore, this kind of signature can only be validated once. However, it is possible to obtain a local copy of the signed quantum state (a known state) [303]. The verification then only invalidates the local copy.

Usage: It is worth noting that to construct a complete user account, the quantum PKI is used differently in quantum blockchains than in classical computing. The access control in quantum blockchains (i.e., ownership of an on-chain state) is assured by interactive signatures on the index of an on-chain state (Section 6.4). Nonetheless, it is also important for the processes of identity attestation and authentication of QBIF (Section 6.4). There are many more research on quantum PKI [299], [300], [301], [302]. For simplicity, Equations (6.1), (6.2), (6.4), and (6.5) are used in

this chapter to respectively represent quantum public-key encryption, decryption, quantum signature, and signature verification.

6.3.2 A Chain of Quantum States

A quantum blockchain can be constructed by a chain of non-tempering and traceable quantum states. Based on quantum entanglement in time, we briefly introduce the scheme of quantum blockchains in [276] as an example. Note that this is a highly simplified version of the classical blockchain. One quantum state constitutes one block. In other words, this design assumes one transaction per block.

Before going to the chain of quantum states, we briefly explain quantum entanglement. The notion of a temper-free chain is inspired by the non-separability of a quantum system (entanglement). Quantum entanglement in time can be viewed as a temporal quantum effect where states are interdependent across time.

If two qubits are entangled, their measurement outcomes are correlated. If we measure one of the qubits, the other qubit instantly collapses into a definite state and would yield a correlated outcome. For example, the entanglement of the Bell states $|x\rangle$ and $|y\rangle$ may be such that the measurement result of $|x\rangle$ is exactly the same as the measurement result of $|y\rangle$. The measurement outcome of them is random, either both being 0 ($|00\rangle$) or both being 1 ($|11\rangle$). Each outcome has a 50% chance.

$$|\psi\rangle = |xy\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}} \qquad (6.6)$$

It is also possible for two Bell states $|x\rangle$ and $|y\rangle$ to be entangled such that the measurement result has a 50:50 chance of being $|01\rangle$ or $|10\rangle$.

Multipartite GHZ (Greenberger-Horne-Zeilinger) states generalize the qubit entanglement to achieve multi-qubit entanglement, in which all the involved states are entangled with each other [304], [305].

Going back to the chain of quantum states, multi-qubit entangled states can be recursively added into a growing chain in chronological order by entanglements in time [276], [306]. This chain of quantum states can be used to constitute a primitive quantum blockchain. For simplicity, a 2-bit string $b_1 b_2$ is used here to represent the data in a block (it can be extended to larger bit strings by multi-qubit entanglement, but still, it depends on the maximum qubit a quantum computer can support). Here, we prepare them into a temporal Bell state:

$$|\psi_{b1b2}\rangle = \frac{|0\rangle\,|b_2\rangle + (-1)^{b_1}\,|1\rangle\,|\overline{b_2}\rangle}{\sqrt{2}} \qquad (6.7)$$

where $|\bar{b}\rangle$ is the negation of b. Now we can create the next temporal Bell state (next block) entangled in time with this state. These two Bell states do not actually coexist, but the second Bell state is correlated with the first one (i.e., the current state is correlated with its previous existing state). Such states were physically generated in the experiment introduced in [306]. Adding subsequent states to them forms a chain of Bell states in chronological order.

Now we add timestamps. A delay line of time T is applied to each Bell state's creation. Each state is marked by its creation time, which becomes the block's timestamp. Let the creation time of the first Bell state (genesis block) be $t = 0$ and the time of the next state (next block) be $t = T$. The genesis block and the next block can be fused into a four-photon GHZ state:

$$|\psi_{b1b2}\rangle^{0,T} = \frac{|0\rangle^{0}\,|b_2\rangle^{T} + (-1)^{b_1}\,|1\rangle^{0}\,|\overline{b_2}\rangle^{T}}{\sqrt{2}} \qquad (6.8)$$

0 is the genesis block's timestamp, and T is the second block's timestamp. Recursively, the blocks become a chain of entangled states (a non-tempering and traceable history of a state). At $t \in \{0, T, 2T, \dots, nT\}$, the chain of quantum states can be represented as

$$|\psi_{b_1 b_2 \dots b_{2n}}\rangle^{0,T,T,2T,2T,3T\cdots,(n-1)T,(n-1)T,nT} = \frac{1}{\sqrt{2}}(|0\rangle^0 |\psi_{b_2}\rangle^T |\psi_{b_3}\rangle^T |\psi_{b_4}\rangle^{2T} \cdots |\psi_{b_{2n-1}}\rangle^{(n-1)T} |\psi_{b_{2n}}\rangle^{nT} +$$

$$(-1)^{b_1}|1\rangle^0|\overline{\psi_{b_2}}\rangle^T|\overline{\psi_{b_3}}\rangle^T|\overline{\psi_{b_4}}\rangle^{2T} \cdots |\overline{\psi_{b_{2n-1}}}\rangle^{(n-1)T}|\overline{\psi_{b_{2n}}}\rangle^{nT}) \qquad (6.9)$$

Each pair of $|\psi_{b_{2i-1}}\rangle^{(i-1)T}|\psi_{b_{2i}}\rangle^{iT}$, $i \in \{2, 3, \dots, 2n\}, b_i \in \{0,1\}$ is a block. $b_{2i-1}b_{2i}$ is the 2-bit data in a block. The superscripts iT represent the block timestamps. For example, suppose the first two blocks contain bit strings 00 and 10, and the first and second blocks are $|\psi_{00}\rangle^{0,T}$ and $|\psi_{10}\rangle^{T,2T}$, respectively. With two blocks, the chain is $|\psi_{0010}\rangle^{0,T,T,2T}$. Adding a third block with value 11 produces $|\psi_{001011}\rangle^{0,T,T,2T,2T,3T}$.

It is worth noting that entanglement in time is distinct from the more commonly used space-based entanglement [307] in which photons in different locations are entangled. The entanglement in time can be viewed as a quantum networked time machine [276].

Besides using GHZ states, quantum blockchains with other kinds of states have also been experimented with. For example, Banerjee et al. create a chain of quantum states using multiparty entanglement of quantum weighted hypergraph states [308]. Bennet et al. propose a quantum-enabled blockchain architecture using quantum information encoded in light (quantum states of light) [309]. Theoretically, the chain of hypergraph states or quantum states of light can also be used in place of the chain of GHZ states and become the data structure of a quantum blockchain using entanglement in time.

Obviously, currently implementable block size and potential chain size are not comparable with the classical blockchains. However, as quantum technologies develop, we believe quantum blockchains can be extended. More limitations and challenges of quantum decentralization are introduced in Section 6.5.

6.3.3 Primitive Consensus

In a quantum blockchain, each blockchain node hosts a copy of the chain of quantum states represented in (6.9). Thus, a consensus protocol is needed to add new blocks securely. It should be able to verify the correctness of the newly added blocks and seek an agreement from all nodes on the validity and sequence of these new blocks. θ-protocol [310] and Quantum Random Number Generator (QRNG) have been proposed to solve such problems. QRNG guarantees the true random selection of a verifier node. True randomness assures fairness and thus should be agreed upon among nodes. The selected verifier node is then responsible for performing a verification test of the new block and adding it to the chain. θ-protocol is then executed to "broadcast" the new block to other parties. In other words, nodes in the blockchain agree on the true randomness instead of solving a cryptographic puzzle or staking coins (e.g., Proof-of-work, Proof-of-Stake). This is similar to leader-based consensus protocols [311]. The fundamental probabilistic property of quantum mechanics gives QRNG the ability to create true randomness, which can be clearly modelized and controlled for perpetual unpredictable randomness, contrary to the predictable randomness generated by classical deterministic processes such as Pseudo-Random Number Generators (PRNG).

Note that the QRNG machine today may have statistical biases due to noisy quantum hardware and is not realistic for use. We assume such bias can be tackled in the future. Nonetheless, the instantly produced randomness may not be able to guarantee simultaneous additions of new blocks.

A complete scheme of consensus mechanisms for quantum blockchains is still under exploration [286], [310]. After the verifier (block producer) selection, the verifier node initiates the verification test (e.g., θ-protocol), which allows each node in the network to verify if a source is distributing a correct state, even in the presence of untrusted nodes. The verifier node finally adds the new block to its local quantum blockchain copy and broadcasts the addition. There are still many limitations to the implementation of quantum broadcasting. More details about broadcasting can be found in Section 6.5. Based on QRNG and θ-protocol, we achieve a primitive consensus process for QBIF, which is discussed next.

6.4 Quantum Blockchain Identity Framework

The vision of QBIF is to achieve secure pseudonymization, which preserves privacy and prevents forgery. Users control their own identities and use them without revealing unnecessary information (self-sovereign identities). At the same time, identities are secure and genuine under a decentralized quantum setting. In this section, we introduce various aspects of the architecture of QBIF: roles, quantum identity attestations, quantum blockchains, authentication, and consensus protocols.

6.4.1 Roles

In this chapter, "identity" includes the user's unique id (e.g., username, social security number), attributes (pieces of information about the user's identity, e.g., name, age, balance, address), and attestations (proofs of user statements about attributes, e.g., "birthyear == 1999," "income > 50,000"). We also use "public credential" to refer to a user's public identity information (unique id and attestations). Identity attributes are private to users (owners). Attribute-based identity

authentication, which verifies identity attributes without revealing additional information, is a prevalent way to authenticate users with anonymity (pseudonymization) [312], [313].

Users can prove that they satisfy the conditions set by identity verifiers without revealing their actual identity information. Identity verifiers authenticate users by verifying the issuers' signatures on the attestations. For example, a bank may loan money only to users whose annual incomes exceed $50,000. When the bank (verifier) evaluates a user's financial information for a loan request, the bank can verify the user's annual income by checking the signature on the user's attestation without looking at the user's actual income number. Keeping attestations on a blockchain ensures their authenticity. Identity attributes are private to owners. Decentralized identity removes the concerns of privacy violation and single-point-of-failure.

As in the classical approach, there are three roles in QBIF: identity owners, identity issuers, and identity verifiers. Identity issuers represent trusted parties such as local governments or big companies. Issuers issue identities such as personal IDs or user profiles in an application/software. They also attest (not issue) the validity of identity attributes and issue attestations. For example, a bank only issues attestations about users' financial information. An attestation process is started on the user's demand and finished by an issuer. The reliability of an attestation depends on the reputation of the issuer. Only attestations are stored on blockchains. Ownership is marked by double signatures, as discussed in Subsection 6.4.5. Users keep their identities at discretion (preferably privately) and use only the on-chain attestations to prove statements about their identity attributes to verifiers. The verifiers provide services according to the reputation of the issuers who attested the identity statements.

In QBIF, attestations are kept in the form of quantum states and are chained together by entanglement. These are called Quantum Identity Attestations (QIAs). Issuers do the generation of QIA. Verifiers on user's demand do authentication of QIA. Identity owners (users) who want authorized actions from verifiers must go through authentication and attestation processes if they have not done that with the issuers.

6.4.2 Quantum Identity Attestations

Attestations are evidence about users' identity statements. They could be in any form (e.g., hash values, encrypted messages, or bit strings). Here, we assume to keep them in bit strings for simplicity. QIAs in QBIF are quantum states containing such bit strings, which could be created by converting them into qubits [314]. Nonetheless, one user can have multiple QIAs (for different identity attributes). Let ψ be the bit string of attestation that consists of n bits; then, a QIA can be represented as

$$L_{\varphi}|0^{\otimes n}\rangle = |\psi\rangle = |b_1 b_2 \cdots b_n\rangle \qquad (6.10)$$

A QIA should be double-signed (by the owner and the issuer) before use, and a signed QIA can be represented as $|\psi\rangle'$ as indicated in (6.11). Signatures are interactively done between an owner and an issuer (signed with private keys from both parties) as discussed in Subsection 6.3.1. $|\psi\rangle'$ has $n + 2(n + m)$ qubits after double-sign, where n is the length of ψ and m is the output length of h defined in (6.3).

$$|\psi\rangle' = |\psi\rangle\, |h^u(\psi, \rho_1^u)\rangle\, |h^i(\psi, h^u(\psi, \rho_1^u), \rho_1^i)\rangle \qquad (6.11)$$

The superscripts indicate which party (u: user, i: issuer) h or ρ_1 belongs to. A QIA is prepared by an issuer per user request in the attestation process. QIAs are double-signed (as $|\psi\rangle'$) and are kept

in the chain of quantum states. Signing with the issuer's private key indicates that the issuer agrees with the user's attested statement. Signing with the user's private key demonstrates the ownership of the attestation. No trusted centralized storage is needed.

6.4.3 QIAs on Quantum Blockchain

We assume a quantum network and achieve decentralization over quantum blockchains. QBIF provides distributed identity management and a primitive consensus protocol for coordinating network nodes (consisting of users, issuers, and verifiers). Nodes are interconnected by quantum communication channels. As shown in Figure 6.2, each party keeps a copy of the chain of quantum states that holds the signed QIAs chronologically. The list of signed QIAs is kept on the chain in the form shown in (6.9).

Issuers and verifiers are candidates for block producers (e.g., miners on proof-of-work blockchains). The owner of a QIA and the issuer must double-sign the QIA before it goes into the block pool, i.e., candidate blocks to be attached to the chain. Block producers add these blocks to the chain after a consensus process (Subsection 6.4.4). The ownership of the signed QIA is demonstrated by the owner's signature, and the validity of the QIA is marked by the issuer's signature. As discussed, the signature could only be done by an interactive signature process. Any QIA signed without the consent of both parties is invalid and does not pass the consensus process.

Once a QIA is put on the chain, the owner can initiate an action with a verifier by sending the location of the signed QIA on the blockchain. The verifier determines the authenticity of the QIA by validating the signed QIA with the user's and the issuer's public keys. No additional knowledge of personal identity is needed during the authentication.

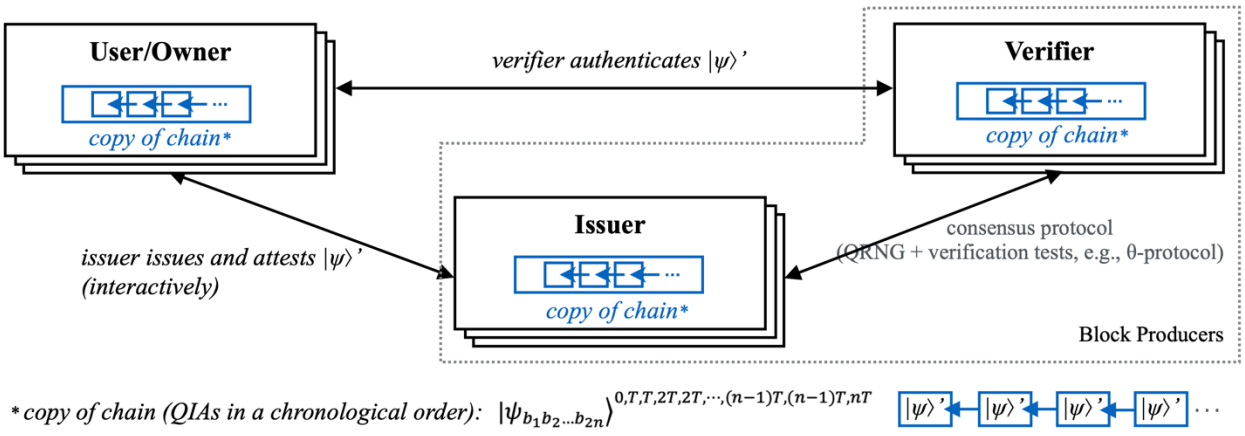


Figure 6.2. *Roles and Interactions on QBIF.*

Note that the level of validity for an attestation can be defined by an issuer so that a user can meet whatever level a verifier may require. The level of validity can be determined by way of issuance or attestation, such as verifying in person, with biometric readers, via password, online, etc. The level of validity can be integrated into QIAs by adding extra qubits describing the level.

Since we are dealing with a primitive quantum blockchain structure, there are no sophisticated event triggers such as those used in smart contracts in classical blockchain-based applications. Thus, part of the communication would be off-chain, such as the online verification mentioned above. A hybrid network of classical and quantum communications would be the most plausible transition from classical internet to quantum internet as quantum technology develops. QKD is one good example.

In QBIF, quantum PKI and quantum blockchain processes are executed on quantum communication channels. Other interactions, such as off-chain attestations and agreements, are kept on classical channels. As shown in Figure 6.3, the network of QBIF is composed of multiple quantum nodes connected via a quantum network in a peer-to-peer fashion. The classical channels are neglected in the figure but coexist with the quantum channels. Each node is a quantum system

hosting a copy of the quantum blockchain, combined with a classical system dealing with off-chain communications.

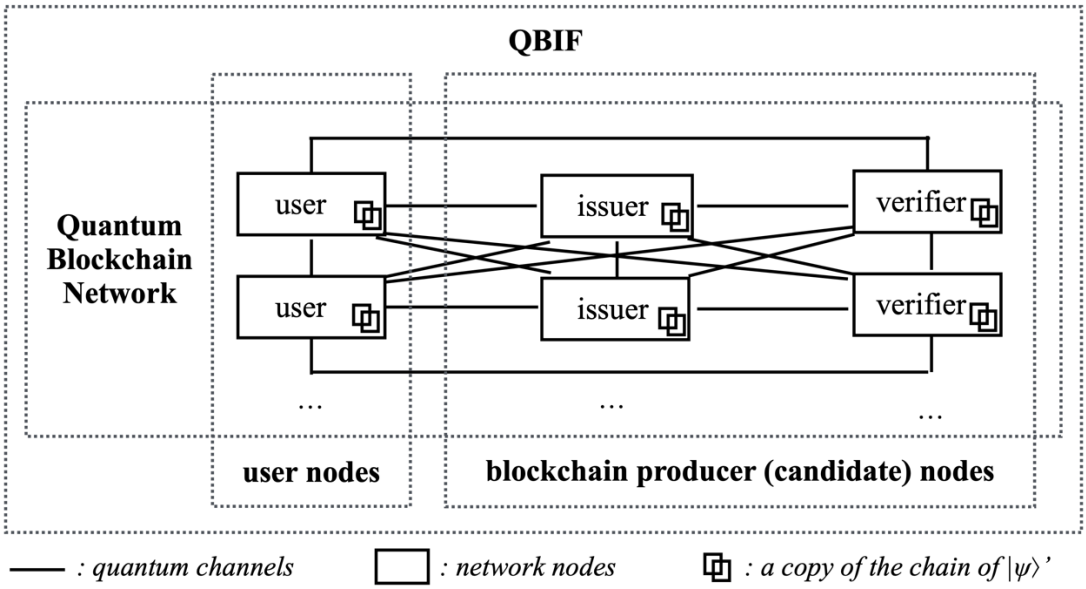


Figure 6.3. *A QBIF Network.*

6.4.4 Consensus Protocol

As discussed earlier, consensus protocols of a quantum blockchain are primitive due to the difficulty of implementing block producers' election (or competition). The most prevalent consensus protocols for classical blockchains are Proof-of-work (mining) in Bitcoin or Proof-of-stake (voting) in Ethereum. Here, we propose to use QRNG and distributed verification tests (e.g., θ-protocol) to emulate an initial consensus protocol for quantum blockchains. QRNG guarantees true randomness in selecting a block producer (who is also responsible for validating a new block). Here, the true randomness of QRNG replaces the fair competition/election provided by Proof-of-work or Proof-of-stake. The selected block producer then executes a distributed verification test to validate the newly added GHZ state (the new block). The distributed verification process keeps the quantum system validated even with untrusted network nodes [310]. Once the verification tests are done, the new block is added to the blockchain. Then, the new chain is broadcasted to all

network nodes. Together with Figure 6.4, the following steps describe the consensus process of QBIF:

1. By now, a double-signed QIA ($|\psi\rangle'$) has been created by an issuer and a user. $|\psi\rangle'$ is waiting to be added to the blockchain. The issuer of $|\psi\rangle'$ initiates the consensus process and activates a QRNG to randomly select a blockchain producer among the candidates.
2. A candidate is selected by the QRNG. If the candidate is unavailable, the QRNG is executed again until an available candidate is selected. We assume the selected candidate is trustworthy (i.e., candidates are trusted). Suppose the 3rd node in Figure 6.4 is available and is selected. It then takes the role of block producer and initiates the distributed verification. For example, in [310], a pass condition and a set of random angles $\theta_j \in [0, \pi)$ are defined in advance, $j \in \{1,2,..,n\}$. θ_j represents the measurement basis of node j, and the verification runs until all the network nodes satisfy the pass condition. For an ideal GHZ state, the verification will succeed theoretically with 100% probability. However, this probability is varied according to entanglement fidelity. $|\psi\rangle'$ is accepted only when the majority of nodes pass the condition [310].
3. Once we have a valid $|\psi\rangle'$, the block producer then fuses $|\psi\rangle'$ into a four-photon GHZ state with the latest on-chain state (6.9), which adds $|\psi\rangle'$ to its local copy of the QIA chain.
4. Then, the block producer "broadcasts" $|\psi\rangle'$ to all network nodes. Note that the verification protocol in [310] involves more than merely implementing verification tests. It also links the outcome of the verification tests, the validated state, to the target state. In other words, the newly added block has also been copied and distributed to all other network nodes during the verification tests. This distribution of the new block remedies the broadcasting limitations of quantum networks. This is discussed further in Section 6.5.

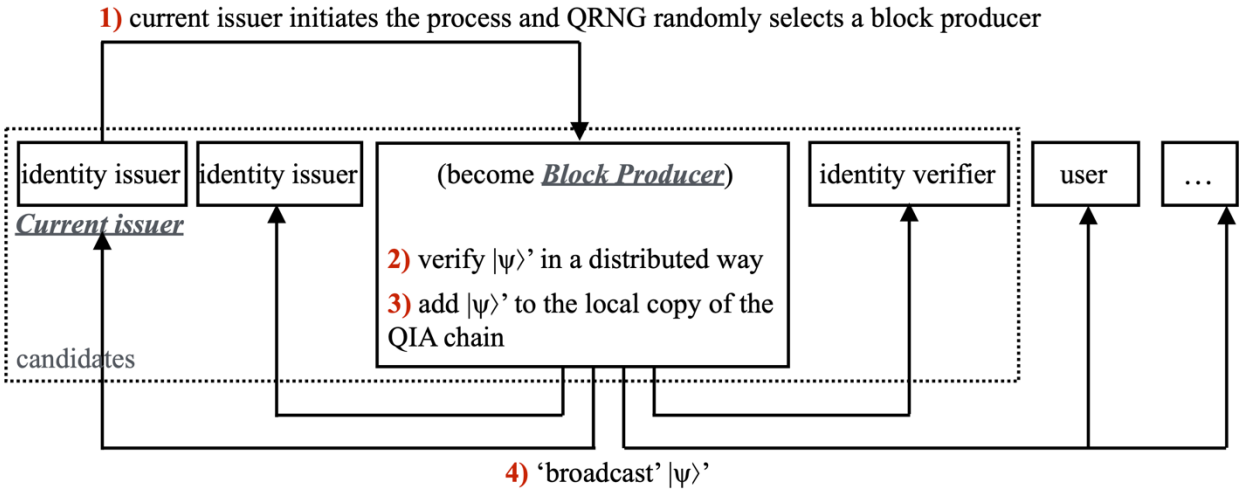


Figure 6.4. *A Consensus Process for Quantum Blockchains.*

6.4.5 Attestation and Authentication

Identity attestation and authentication are two critical processes of QBIF. The attestation process produces QIAs, and the authentication process uses QIAs. In Figures 6.5 and 6.7, dashed arrow lines refer to classical communication channels, and solid arrow lines represent quantum channels.

Attestation process: The attestation process is completed by collaboration between the user and an issuer. This process creates QIAs and makes interactive signatures. Figure 6.5 shows a step-by-step attestation process. Together with Figure 6.5, the following steps describe the identity attestation process:

1. A user initiates the process by requesting an identity attestation. The user defines the statements about the identity attributes (e.g., income > 50,000). The issuer validates the user's identity attributes. This step can be done over a classical communication channel.
2. The issuer prepares the QIA $|\psi\rangle$ (6.10) and sends it to the user over a quantum communication channel.
3. The user receives $|\psi\rangle$ and starts the interactive signature (as discussed in Subsection 6.3.1) with the issuer. $|\psi\rangle$ is first interactively signed with the user's private key and then with

the issuer's private key (6.11). The interactive signatures prove the ownership and the validity of the QIA. After $|\psi\rangle$ is signed from both sides, $|\psi\rangle'$ is ready.

4. The issuer initiates the consensus process by activating the QRNG.
5. The selected block producer starts the distributed verification tests. If the tests are passed, $|\psi\rangle'$ is added to the QIA chain and is broadcasted to the network.
6. The block producer sends the location nT of $|\psi\rangle'$ in the blockchain to the user using a classical communication channel. Since there is no available Decentralized Identifier (DID) to identify users in QBIF (which is generally used by blockchain-based identity), the user keeps the signed QIA's location. It is shown to the verifier in the authentication process and used to localize the user's signed QIA. The user can claim ownership of this signed QIA by private key.

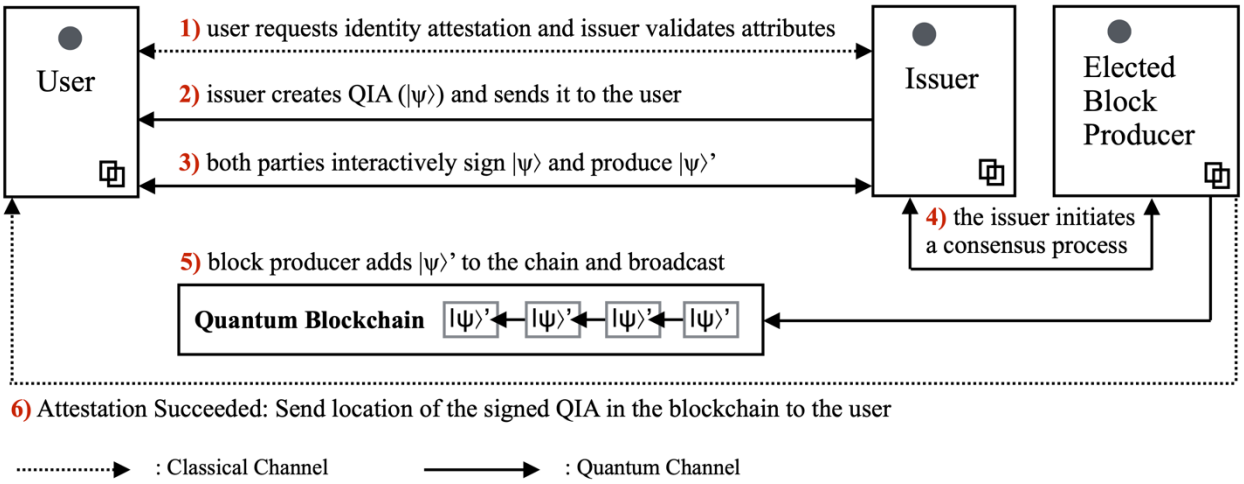


Figure 6.5. *An Attestation Process for QBIF.*

Authentication process: The authentication process happens between a user and a verifier when the user requests an authorized action. It requires a signed on-chain QIA corresponding to the action's requirements. The verifier extracts the signed QIA and uses it to authenticate the user. As shown in Figure 6.7, the following steps describe the authentication process:

1. A user requests an authorized action provided by a verifier, which initiates the authentication process. The verifier verifies whether the user satisfies the requirement (e.g., if the user's annual income exceeds $50,000 when applying for a loan).
2. The user interactively signs the QIA location nT (as $|nT\rangle\,|f(nT,\rho_1^{u\text{-}l})\rangle$) with the verifier using the same private key as the one used for signing the QIA and then sends it to the verifier. This signature of location nT proves the user's ownership of the private key used to sign the QIA. As discussed in Subsection 6.3.1, only an interactive signature process can guarantee a signature's validity (not someone pretending to own this signature). Therefore, the user must interactively sign and send the QIA location to the verifier through a quantum channel.
3. The user announces his/her public key if necessary.
4. The verifier extracts the identity location nT according to (6.5). By nT, the verifier finds and obtains the signed QIA $|\psi\rangle'$.
5. The verifier verifies the signatures of $|\psi\rangle'$ with the issuer and the user's public key. If the signature is valid, the verifier uses ψ to determine the authentication result. Based on (6.5) and (6.11), the extraction and verification of $|\psi\rangle'$ is described in Algorithm 6.1 (Figure 6.6).
6. If the process succeeds, the verifier makes authenticated communications with the user (Section 6.3.1) and proceeds with the requested service. (e.g., the verifier successfully verifies that the user's income exceeds $50,000 and proceeds with the subsequent loan application procedures.)

Algorithm 1. Extract and verify a QIA

Quantum transformation: U^{evqia}

Input: $|\psi\rangle', |nT\rangle\ |f(nT, \rho_1^{u_l})\rangle, h, \rho_1^u, \rho_2^u, \rho_1^{u_l}, \rho_2^{u_l}$

#Superscript denotation: (u: user, i: issuer, u_l: QIA location).

#Extract $|\psi\rangle'$ by the issuer's public key

$|\psi\rangle' = |\psi\rangle\ |h^u(\psi, \rho_1^u)\rangle\ |h^i(\psi, h^u(\psi, \rho_1^u), \rho_1^i)\rangle \rightarrow |\psi\rangle\ |h^u(\psi, \rho_1^u)\rangle\ |0\rangle$

#Extract $|\psi\rangle\ |h^u(\psi, \rho_1^u)\rangle\ |0\rangle$ by the user's public key

$(\rho_r^{u'}, \rho_s^{u'}) = h(\rho_1^u, \rho_2^u)$

if $\rho_r^u == \rho_r^{u'}$ && $M(|\psi\rangle\ |h^u(\psi, \rho_1^u)\rangle\ |0\rangle, index_{last})$ *is* $|0\rangle$

then

#Extract $|nT\rangle\ |f(nT, \rho_1^{u_l})\rangle$ by the user's public key and verify signatures

$|nT\rangle\ |f(nT, \rho_1^{u_l})\rangle \rightarrow |nT\rangle\ |0\rangle$

$(\rho_r^{u_l'}, \rho_s^{u'}) = h(\rho_1^{u_l}, \rho_2^{u_l})$

if $\rho_r^{u_l} == \rho_r^{u_l'}$ && $M(|nT\rangle\ |0\rangle, index_{last})$ *is* $|0\rangle$ && $\rho_1^{u_l} == \rho_1^u$

then

$|\psi\rangle\ |h^u(\psi, \rho_1^u)\rangle\ |0\rangle \rightarrow |\psi\rangle\ |h^u(\psi, \rho_1^u)\rangle \rightarrow |\psi\rangle\ |0\rangle \rightarrow |\psi\rangle$

return $|\psi\rangle$ #Succeed

end if

end if

Figure 6.6. *Algorithm 6.1. Extracting and Verifying a QIA.*

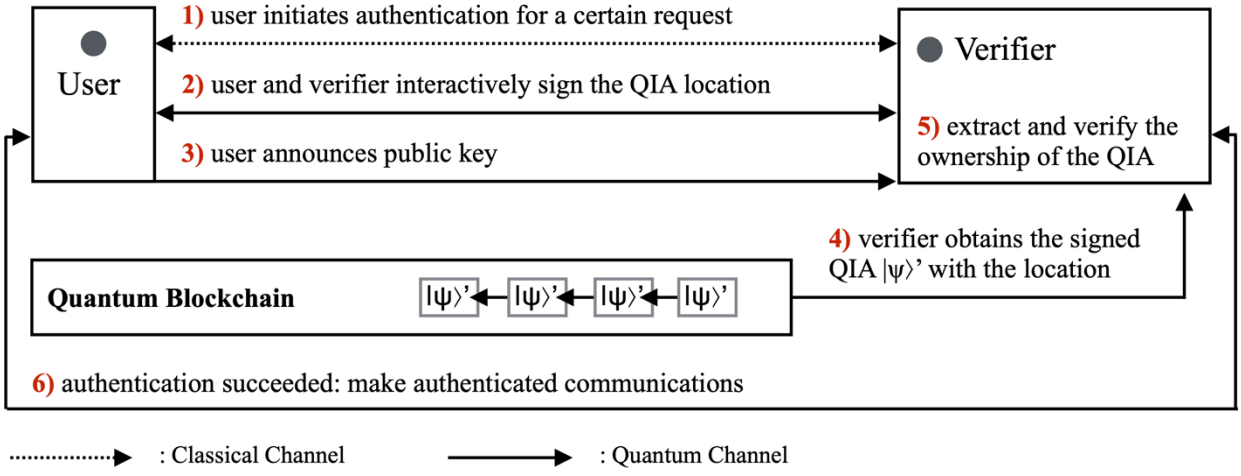


Figure 6.7. *An Authentication Process for QBIF.*

6.4.6 System Security

The security of QBIF is mainly based on the quantum trapdoor one-way functions (6.1) and the secure nature of quantum physics ("unconditional security"). Here, we discuss the security aspects of QBIF:

Quantum-proof: As discussed, classical blockchains' security is vulnerable to quantum computing. Quantum blockchains are immune to any notable quantum and classical attack.

Data integrity: Classical blockchains use hash functions to chain blocks (and Merkle trees to organize transactions) and thus are sensitive to data tampering. If a random on-chain block is tampered with, all its following blocks become invalid because each block points to its previous block. The link of hash values is broken from the tampered block to the latest block. For quantum blockchains, such sensitivity has been kept and amplified by entanglement. One change to a single block could demolish the whole copy of the state chain.

Anti-spoofing: Assume that an attacker wants to perform a spoofing attack, masquerading as a user to claim the user's on-chain attestations. The attacker's goal is to pass an authentication process fraudulently. Suppose the attacker has infiltrated the quantum channel between the user and the verifier (Steps 2 and 3 in Figure 6.7). The attacker can now intercept the communication of the QIA location. However, the attacker still needs to interactively sign it with the verifier. If the attacker uses his/her own private key and sends his/her public key, the verifier is not able to extract the signed QIA by this public key and ρ_1 (6.4) (6.5). $\rho_1^{attacker}$ of the signed QIA location is not the same as ρ_1^{user} of the signed QIA.

Privacy: On-chain attestations are proofs of statements about users' identity attributes without actual identity information (pseudonymization). Such granular statements would not reveal much information, even if leaked. Pseudonymization protects privacy through the non-correlation principles [312].

6.5 Challenges and Limitations

Quantum blockchains are naturally immune to quantum attacks on classical algorithms. It is exciting to see the possibility of layering blockchains on a quantum system, even though it is not pure quantum (but a hybrid with classical systems). In the QBIF conceptual design, every preliminary technology has been explicitly shown to be experimentally realizable, even though some are only in their simplest form. The proposed conceptual design does not yet meet immediate deployment requirements, but it shows a visible path toward realizing a decentralized quantum application. Nonetheless, many challenges and limitations of quantum approaches are inevitable at this stage. We discuss the main challenges and limitations of QBIF next.

Lack of implementation: QBIF and most quantum blockchains lack low-level specifics, which would contribute to the detailed implementation of their essential components. Implementations of many quantum technologies are still varying and require much further experimentation. Quantum blockchains are still in their very early phase. Quantum systems are mostly implemented in simulators or quantum computers with limited resources. It would not add much value to the work at this stage by introducing implementation details.

Limited quantum hardware: The development of quantum computers has been rapid in recent years. However, it still needs a significant time before it can support large or medium-scale computation (and connection). Current quantum computers are extremely sensitive to interactions; thus, their computation results usually have much noise. Besides, quantum states tend to gradually disintegrate due to quantum decoherence, which demands quantum error correction. State discrimination strategies must be applied to fine-tune the trade-offs between error rate and

information gain [271]. In terms of authentication, the validation of a whole process would be determined or significantly influenced by the quantum bit error rate [285].

Broadcast limitations: The ability to distribute entangled states to network nodes is significant for a decentralized quantum network. Theoretical analyses on point-to-point quantum communication have constantly been progressing. Quantum repeaters make possible long-distance quantum communications over bipartite quantum channels. However, point-to-point communication is not enough for a sophisticated network. A quantum broadcast network would be able to distribute entangled qubits efficiently. Current quantum broadcast networks are usually accomplished with the assistance of classical communications [315]. The upper and lower bounds of quantum broadcast networks are still limited [316], and a complex network typically introduces more noise and losses.

Temporal entanglement: There are still many open problems with quantum entanglement in time [276], [317]. Entanglement in time is based on entangled history theory, which is a substantial modification of the decoherent history theory (or consistent history theory). Entanglement in time shares similar properties with the general entanglement in space [307], but temporal quantum effects might be highly unstable as time passes.

Difficulty in accessing states: Previous blocks on quantum blockchains based on entanglement in time are physically no longer existent. Blocks do not simultaneously coexist. It is only natural to retrieve the latest block. Previous on-chain states are hard to access. However, it is possible to transfer the past quantum correlation between quantum states so that past states can be obtained, even if they do not coexist simultaneously [303]. Moreover, quantum logic is inherently reversible; thus, propagating back in time to a previous block will unentangle qubits and potentially destroy

the chain. Fortunately, through the consensus process, we have many copies of the chain on the blockchain nodes. Hence, the system is feasible as long as the majority of the blockchain nodes still keep the copies. How to make the system more sustainable than this remains future work.

Overhead: Network overhead is significantly increased since signatures are done interactively between two parties. Signature processes are common in blockchains. Moreover, many quantum signatures can only be validated once. This would significantly increase the number of signature processes and potentially overload the network nodes. Quantum fingerprints could replace such interactive signatures to void this issue.

Scalability: Scalability is always an issue in quantum applications due to the current scale of quantum computers and networks. In terms of quantum blockchains, one transaction per block is clearly deficient. A complex quantum data structure must be developed before a quantum blockchain can become comparable to its classical counterpart.

Primitive consensus: The current consensus protocol is still primitive, lacking perfect competition/election rules to create blocks. QRNG is temporarily used as a basic consensus protocol. However, QRNG cannot guarantee the normal operation of simultaneous communications in a decentralized network.

Low efficiency: Performance issues of quantum blockchains exist in the information preparation [314], [318], block size [276], and cost-effectiveness strategies between the used qubit and the created block [308]. Blockchain size is a challenge for classical blockchains also due to data explosion.

Semi-decentralization: The classical identity management and this approach require third-party authorities to verify user attestations and thus are not actual decentralization but semi-

decentralization. The idea is similar to what a consortium blockchain provides, in which each member is a group of nodes who believe in the same authorities [319]. A semi-decentralization could lead to the dictatorship of resources by a small group of powerful entities. However, we believe a proper combination of decentralization and centralization is beneficial since there are always harmful contents that could inundate the network without censorship. How an ultimate decentralization can be achieved in the quantum domain (or in any future form of the internet) remains future work.

6.6 Summary

Quantum computers threaten classical blockchains' security, and thus quantum blockchains are needed. This chapter theoretically analyzes the quantum blockchain technologies that could be used for a decentralized purpose. We review the technical evidence and put together a conceptual design for decentralized quantum identity, which is named QBIF. We investigate the essential components and the feasibility, effectiveness, and limitations of QBIF. We complete the design of QBIF by integrating existing quantum approaches, such as quantum PKIs and quantum blockchains. Nonetheless, the conceptual design still has technical gaps in detailed low-level specifics. Despite the limitations and challenges, we believe a decentralized perspective of quantum applications is noteworthy and likely.

Chapter 7: Conclusion

This chapter concludes this dissertation with future directions.

7.1 Summary of Dissertation

Despite the promising advantages quantum computers can offer, the capabilities of individual devices remain limited, and monolithic hardware scaling poses substantial challenges. This dissertation addresses these scalability issues across the domains of quantum networks, DQC, and security, with the overarching goal of improving the performance, scalability, practicality, and security of quantum systems.

Chapter 1 introduces the foundational components and security considerations of quantum networks and DQC, along with the motivations, goals, and contributions of this dissertation.

In Chapter 2, we propose AER to enhance the performance and practicality of quantum routing by removing the need for synchronized operations and increasing entanglement utility. AER operates in a fully distributed manner using only local neighbor information and structures like DODAGs or spanning trees. By mitigating delays from global network updates, AER significantly improves E2E entanglement efficiency. Our results show that AER outperforms synchronous approaches in both entanglement rate and scalability, offering a promising direction for future quantum networks.

Chapter 3 extends AER into a multi-tree routing scheme to support larger-scale quantum networks. This method connects multiple DODAGs in a distributed, asynchronous fashion while relying solely on local entanglement link information. Simulations across both synthetic and realistic topologies show that the multi-tree approach achieves higher entanglement rates and greater

efficiency than single-tree and synchronous routing strategies, demonstrating the effectiveness of asynchronous methods for diverse network architectures.

In Chapter 4, we shift focus from connecting external systems to linking internal components, addressing the growing need for scalable DQC architectures. In quantum data centers with multiple QPUs across different racks, executing remote gates introduces resource contention. To tackle this, we propose a resource management framework that maximizes fidelity-guaranteed throughput while respecting gate dependencies. The problem is formulated as a MILP model for benchmarking, and we develop efficient heuristic scheduling algorithms that closely approximate the MILP results with significantly reduced computation, making them practical for real-world DQC scenarios.

Chapter 5 focuses on the security of quantum networks. While quantum communication offers inherent security advantages over classical systems, the quantum internet still faces unique challenges in ensuring CIA. We analyze vulnerabilities and mitigation strategies across the physical, link, network, and application layers. Our evaluation integrates both classical and quantum techniques to assess attack severity and the effectiveness of countermeasures, emphasizing the need for adaptive, cross-layer defenses to ensure robust and trustworthy quantum communication.

Finally, in Chapter 6, we explore quantum-era cryptographic applications through a theoretical analysis of quantum blockchains for decentralized identity authentication. We propose a conceptual framework, review the supporting technical literature, and assess its key components in terms of feasibility, effectiveness, and limitations.

7.2 Future Directions

While the proposed solutions in this dissertation improve practicality, performance, and resilience, several open challenges remain, pointing to promising avenues for future research.

Cross-Layer Co-Design of Quantum Network Architectures: A key research direction lies in developing a comprehensive cross-layer design that jointly considers physical parameters (e.g., fidelity, decoherence), network-layer routing, and application-level requirements (e.g., latency, throughput). Co-designing protocols across layers can lead to improved resource allocation and performance guarantees. Machine learning and reinforcement learning could also play a role in enabling adaptive decision-making under real-world network dynamics; however, their reliance on classical processing may introduce overhead that is impractical given the short-lived nature of quantum states in near-term technologies. Thus, a balanced trade-off strategy is also necessary.

Scalable Resource Orchestration: Resource management for DQC provides a foundation for optimizing fidelity and execution time in multi-QPU systems. This can be extended by incorporating heterogeneous hardware models, shared memory architectures, and real-time feedback for dynamic scheduling. Additionally, exploring integration with quantum-classical hybrid cloud environments could open new possibilities for scaling practical quantum workloads.

Security Frameworks for Hybrid Quantum-Classical Systems: Quantum networks will likely operate alongside classical infrastructure for the foreseeable future. Developing secure protocols that bridge both quantum and classical domains, including authentication, integrity verification, and fault isolation, is critical. Future work could focus on building modular, adaptive security architectures capable of responding to dynamic network behavior and supporting a broader range of quantum cryptographic primitives.

References

[1] Z. Yang, M. Zolanvari, and R. Jain, "A Survey of Important Issues in Quantum Computing and Communications," *IEEE Commun. Surv. Tutor.*, vol. 25, no. 2, pp. 1059–1094, 2023, doi: 10.1109/COMST.2023.3254481.

[2] N. Gisin and R. Thew, "Quantum communication," *Nat. Photonics*, vol. 1, no. 3, pp. 165–171, Mar. 2007, doi: 10.1038/nphoton.2007.22.

[3] A. K. Ekert, "Quantum cryptography based on Bell's theorem," *Phys. Rev. Lett.*, vol. 67, no. 6, pp. 661–663, Aug. 1991, doi: 10.1103/PhysRevLett.67.661.

[4] C. L. Degen, F. Reinhard, and P. Cappellaro, "Quantum sensing," *Rev. Mod. Phys.*, vol. 89, no. 3, p. 035002, Jul. 2017, doi: 10.1103/RevModPhys.89.035002.

[5] B. A. Cordier, N. P. D. Sawaya, G. G. Guerreschi, and S. K. McWeeney, "Biology and medicine in the landscape of quantum advantages," *J. R. Soc. Interface*, Nov. 2022, doi: 10.1098/rsif.2022.0541.

[6] Y. Cao *et al.*, "Quantum Chemistry in the Age of Quantum Computing," *Chem. Rev.*, Aug. 2019, doi: 10.1021/acs.chemrev.8b00803.

[7] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*, Anniversary edition. Cambridge ; New York: Cambridge University Press, 2011.

[8] E. G. Rieffel and W. H. Polak, *Quantum Computing: A Gentle Introduction*, Illustrated edition. Cambridge, Massachusetts London, England: The MIT Press, 2014.

[9] P. Kaye, R. Laflamme, and M. Mosca, *An Introduction to Quantum Computing*, 1st edition. Oxford: Oxford University Press, 2007.

[10] J. Preskill, "Quantum computing in the NISQ era and beyond," *Quantum*, vol. 2, p. 79, Aug. 2018, doi: 10.22331/q-2018-08-06-79.

[11] A. P. Vepsäläinen *et al.*, "Impact of ionizing radiation on superconducting qubit coherence," *Nature*, vol. 584, no. 7822, pp. 551–556, Aug. 2020, doi: 10.1038/s41586-020-2619-8.

[12] M. I. Dyakonov, "Is fault-tolerant quantum computation really possible," *Future Trends Microelectron. Nano Creek*, vol. 4, 2007, Accessed: Mar. 09, 2025. [Online]. Available: https://books.google.com/books?hl=en&lr=&id=J1_4DwAAQBAJ&oi=fnd&pg=PA4&dq=Is+Fault-Tolerant+Quantum+Computation+Really+Possible&ots=a2Pqw8fMAv&sig=WPLeYJBqRxrBmDLXLqdQyiSpLC8

[13] Nu Quantum, "Distributed Quantum Error Correction: theory breakthrough from Nu Quantum charts pathway for quantum computing scale-out." Accessed: Mar. 09, 2025. [Online]. Available: https://www.nu-quantum.com/news/distributed-quantum-error-correction-theory-breakthrough-from-nu-quantum-charts-pathway-for-quantum-computing-scale-out

[14] M. Caleffi, M. Amoretti, D. Ferrari, J. Illiano, A. Manzalini, and A. S. Cacciapuoti, "Distributed quantum computing: A survey," *Comput. Netw.*, vol. 254, p. 110672, Dec. 2024, doi: 10.1016/j.comnet.2024.110672.

[15] H. J. Kimble, "The quantum internet," *Nature*, vol. 453, no. 7198, pp. 1023–1030, Jun. 2008, doi: 10.1038/nature07127.

[16] W. Kozlowski *et al.*, "Architectural Principles for a Quantum Internet," Internet Engineering Task Force, Request for Comments RFC 9340, Mar. 2023. doi: 10.17487/RFC9340.

[17] Z. Li *et al.*, "Entanglement-assisted quantum networks: Mechanics, enabling technologies, challenges, and research directions," *IEEE Commun. Surv. Tutor.*, 2023.

[18] J. Borregaard, H. Pichler, T. Schröder, M. D. Lukin, P. Lodahl, and A. S. Sørensen, "One-Way Quantum Repeater Based on Near-Deterministic Photon-Emitter Interfaces," *Phys. Rev. X*, vol. 10, no. 2, p. 021071, Jun. 2020, doi: 10.1103/PhysRevX.10.021071.

[19] H. Buhrman and H. Röhrig, "Distributed Quantum Computing," in *Mathematical Foundations of Computer Science 2003*, B. Rovan and P. Vojtáš, Eds., Berlin, Heidelberg: Springer, 2003, pp. 1–20. doi: 10.1007/978-3-540-45138-9_1.

[20] R. Van Meter and S. J. Devitt, "The path to scalable distributed quantum computing," *Computer*, vol. 49, no. 9, pp. 31–42, Sep. 2016, doi: 10.1109/MC.2016.291.

[21] H. J. Briegel, W. Dür, J. I. Cirac, and P. Zoller, "Quantum repeaters: the role of imperfect local operations in quantum communication," *Phys. Rev. Lett.*, vol. 81, no. 26, pp. 5932–5935, Dec. 1998, doi: 10.1103/PhysRevLett.81.5932.

[22] K. Azuma, K. Tamaki, and H.-K. Lo, "All-photonic quantum repeaters," *Nat. Commun.*, vol. 6, no. 1, Art. no. 1, Apr. 2015, doi: 10.1038/ncomms7787.

[23] B. Hensen *et al.*, "Loophole-free Bell inequality violation using electron spins separated by 1.3 kilometres," *Nature*, vol. 526, no. 7575, pp. 682–686, Oct. 2015, doi: 10.1038/nature15759.

[24] L. Gyongyosi, "Dynamics of entangled networks of the quantum Internet," *Sci. Rep.*, vol. 10, no. 1, p. 12909, Jul. 2020, doi: 10.1038/s41598-020-68498-x.

[25] A. Pirker, J. Wallnöfer, and W. Dür, "Modular architectures for quantum networks," *New J. Phys.*, vol. 20, no. 5, p. 053054, May 2018, doi: 10.1088/1367-2630/aac2aa.

[26] J. Illiano, M. Caleffi, A. Manzalini, and A. S. Cacciapuoti, "Quantum Internet protocol stack: A comprehensive survey," *Comput. Netw.*, vol. 213, p. 109092, Aug. 2022, doi: 10.1016/j.comnet.2022.109092.

[27] T. Satoh, S. Nagayama, S. Suzuki, T. Matsuo, M. Hajdušek, and R. V. Meter, "Attacking the Quantum Internet," *IEEE Trans. Quantum Eng.*, vol. 2, pp. 1–17, 2021, doi: 10.1109/TQE.2021.3094983.

[28] D. Bruß, "Optimal Eavesdropping in Quantum Cryptography with Six States," *Phys. Rev. Lett.*, vol. 81, no. 14, pp. 3018–3021, Oct. 1998, doi: 10.1103/PhysRevLett.81.3018.

[29] H.-K. Lo and S. Popescu, "Classical Communication Cost of Entanglement Manipulation: Is Entanglement an Interconvertible Resource?," *Phys. Rev. Lett.*, vol. 83, no. 7, pp. 1459–1462, Aug. 1999, doi: 10.1103/PhysRevLett.83.1459.

[30] S. Wehner, D. Elkouss, and R. Hanson, “Quantum internet: A vision for the road ahead,” *Science*, vol. 362, no. 6412, p. eaam9288, Oct. 2018, doi: 10.1126/science.aam9288.

[31] F. Hahn, A. Pappa, and J. Eisert, “Quantum network routing and local complementation,” *Npj Quantum Inf.*, vol. 5, no. 1, pp. 1–7, Sep. 2019, doi: 10.1038/s41534-019-0191-6.

[32] D. Cuomo, “Architectures and circuits for distributed quantum computing,” Jul. 16, 2023, *arXiv*: arXiv:2307.07908. doi: 10.48550/arXiv.2307.07908.

[33] Y. Makhlin, G. Schön, and A. Shnirman, “Quantum-state engineering with Josephson-junction devices,” *Rev. Mod. Phys.*, vol. 73, no. 2, pp. 357–400, May 2001, doi: 10.1103/RevModPhys.73.357.

[34] “IBM Launches Its Most Advanced Quantum Computers, Fueling New Scientific Value and Progress towards Quantum Advantage,” IBM Newsroom. Accessed: Mar. 09, 2025. [Online]. Available: https://newsroom.ibm.com/2024-11-13-ibm-launches-its-most-advanced-quantum-computers,-fueling-new-scientific-value-and-progress-towards-quantum-advantage

[35] J. Gambetta, “The hardware and software for the era of quantum utility is here.” Accessed: Jul. 03, 2024. [Online]. Available: https://www.ibm.com/quantum/blog/quantum-roadmap-2033

[36] H. Neven, “Meet Willow, our state-of-the-art quantum chip,” Google. Accessed: Dec. 21, 2024. [Online]. Available: https://blog.google/technology/research/google-willow-quantum-chip/

[37] S. Slussarenko and G. J. Pryde, “Photonic quantum information processing: A concise review,” *Appl. Phys. Rev.*, vol. 6, no. 4, p. 041303, Oct. 2019, doi: 10.1063/1.5115814.

[38] “Xanadu | X-Series.” Accessed: Mar. 09, 2025. [Online]. Available: https://www.xanadu.ai/products/x-series

[39] PsiQuantum, “A fast path to useful, error-corrected quantum computing.” Accessed: Mar. 09, 2025. [Online]. Available: https://www.psiquantum.com/blueprint

[40] C. D. Bruzewicz, J. Chiaverini, R. McConnell, and J. M. Sage, “Trapped-ion quantum computing: Progress and challenges,” *Appl. Phys. Rev.*, vol. 6, no. 2, p. 021314, May 2019, doi: 10.1063/1.5088164.

[41] “IonQ Tempo - A Paradigm Shifting Quantum Computer,” IonQ. Accessed: Mar. 09, 2025. [Online]. Available: https://ionq.com/quantum-systems/tempo

[42] A. Stern, “Anyons and the quantum Hall effect—A pedagogical review,” *Ann. Phys.*, vol. 323, no. 1, pp. 204–249, Jan. 2008, doi: 10.1016/j.aop.2007.10.008.

[43] M. Cheng, R. M. Lutchyn, and S. Das Sarma, “Topological protection of Majorana qubits,” *Phys. Rev. B*, vol. 85, no. 16, p. 165124, Apr. 2012, doi: 10.1103/PhysRevB.85.165124.

[44] L. Jiang, C. L. Kane, and J. Preskill, “Interface between Topological and Superconducting Qubits,” *Phys. Rev. Lett.*, vol. 106, no. 13, p. 130504, Mar. 2011, doi: 10.1103/PhysRevLett.106.130504.

[45] “Microsoft’s Majorana 1 chip carves new path for quantum computing,” Source. Accessed: Mar. 09, 2025. [Online]. Available:

https://news.microsoft.com/source/features/innovation/microsofts-majorana-1-chip-carves-new-path-for-quantum-computing/

[46] W. K. Wootters and W. H. Zurek, “A single quantum cannot be cloned,” *Nature*, vol. 299, no. 5886, Art. no. 5886, Oct. 1982, doi: 10.1038/299802a0.

[47] J. S. Sidhu *et al.*, “Advances in space quantum communications,” *IET Quantum Commun.*, vol. 2, no. 4, pp. 182–217, 2021, doi: 10.1049/qtc2.12015.

[48] D. Cozzolino, B. Da Lio, D. Bacco, and L. K. Oxenløwe, “High-Dimensional Quantum Communication: Benefits, Progress, and Future Challenges,” *Adv. Quantum Technol.*, vol. 2, no. 12, p. 1900038, 2019, doi: 10.1002/qute.201900038.

[49] C. H. Bennett and G. Brassard, “Quantum cryptography: Public key distribution and coin tossing,” *Theor. Comput. Sci.*, vol. 560, pp. 7–11, Dec. 2014, doi: 10.1016/j.tcs.2014.05.025.

[50] M. Lucamarini, Z. L. Yuan, J. F. Dynes, and A. J. Shields, “Overcoming the rate–distance limit of quantum key distribution without quantum repeaters,” *Nature*, vol. 557, no. 7705, pp. 400–403, May 2018, doi: 10.1038/s41586-018-0066-6.

[51] R. J. Hughes, Morgan ,George L., and C. G. and Peterson, “Quantum key distribution over a 48 km optical fibre network,” *J. Mod. Opt.*, vol. 47, no. 2–3, pp. 533–547, Feb. 2000, doi: 10.1080/09500340008244058.

[52] P. C. Humphreys *et al.*, “Deterministic delivery of remote entanglement on a quantum network,” *Nature*, vol. 558, no. 7709, pp. 268–273, Jun. 2018, doi: 10.1038/s41586-018-0200-5.

[53] K. Wang *et al.*, “Entangling Independent Particles by Path Identity,” *Phys. Rev. Lett.*, vol. 133, no. 23, p. 233601, Dec. 2024, doi: 10.1103/PhysRevLett.133.233601.

[54] E. Knill and R. Laflamme, “Theory of quantum error-correcting codes,” *Phys. Rev. A*, vol. 55, no. 2, pp. 900–911, Feb. 1997, doi: 10.1103/PhysRevA.55.900.

[55] J. Schupp, V. Krcmarsky, V. Krutyanskiy, M. Meraner, T. E. Northup, and B. P. Lanyon, “Interface between Trapped-Ion Qubits and Traveling Photons with Close-to-Optimal Efficiency,” *PRX Quantum*, vol. 2, no. 2, p. 020331, Jun. 2021, doi: 10.1103/PRXQuantum.2.020331.

[56] P. Kobel, M. Breyer, and M. Köhl, “Deterministic spin-photon entanglement from a trapped ion in a fiber Fabry–Perot cavity,” *Npj Quantum Inf.*, vol. 7, no. 1, pp. 1–7, Jan. 2021, doi: 10.1038/s41534-020-00338-2.

[57] V. Krutyanskiy *et al.*, “Entanglement of Trapped-Ion Qubits Separated by 230 Meters,” *Phys. Rev. Lett.*, vol. 130, no. 5, p. 050803, Feb. 2023, doi: 10.1103/PhysRevLett.130.050803.

[58] J. Hannegan, J. D. Siverns, and Q. Quraishi, “Entanglement between a trapped-ion qubit and a 780-nm photon via quantum frequency conversion,” *Phys. Rev. A*, vol. 106, no. 4, p. 042441, Oct. 2022, doi: 10.1103/PhysRevA.106.042441.

[59] B. M. Terhal, “Quantum supremacy, here we come,” *Nat. Phys.*, vol. 14, no. 6, pp. 530–531, Jun. 2018, doi: 10.1038/s41567-018-0131-y.

[60] S. Boixo *et al.*, “Characterizing quantum supremacy in near-term devices,” *Nat. Phys.*, vol. 14, no. 6, pp. 595–600, Jun. 2018, doi: 10.1038/s41567-018-0124-x.

[61] J. C. Chapman and N. A. Peters, “Paving the Way for Satellite Quantum Communications,” *Physics*, vol. 15, p. 172, Nov. 2022, doi: 10.1103/RevModPhys.94.035001.

[62] G. Vallone *et al.*, “Experimental Satellite Quantum Communications,” *Phys. Rev. Lett.*, vol. 115, no. 4, p. 040502, Jul. 2015, doi: 10.1103/PhysRevLett.115.040502.

[63] M. Aspelmeyer, T. Jennewein, M. Pfennigbauer, W. R. Leeb, and A. Zeilinger, “Long-distance quantum communication with entangled photons using satellites,” *IEEE J. Sel. Top. Quantum Electron.*, vol. 9, no. 6, pp. 1541–1551, Nov. 2003, doi: 10.1109/JSTQE.2003.820918.

[64] Y.-A. Chen *et al.*, “An integrated space-to-ground quantum communication network over 4,600 kilometres,” *Nature*, vol. 589, no. 7841, pp. 214–219, Jan. 2021, doi: 10.1038/s41586-020-03093-8.

[65] J. Yin *et al.*, “Satellite-based entanglement distribution over 1200 kilometers,” *Science*, vol. 356, no. 6343, pp. 1140–1144, Jun. 2017, doi: 10.1126/science.aan3211.

[66] D. Niu, Y. Zhang, A. Shabani, and H. Shapourian, “All-photonic one-way quantum repeaters with measurement-based error correction,” *Npj Quantum Inf.*, vol. 9, no. 1, pp. 1–9, Oct. 2023, doi: 10.1038/s41534-023-00775-9.

[67] S. Muralidharan, L. Li, J. Kim, N. Lütkenhaus, M. D. Lukin, and L. Jiang, “Optimal architectures for long distance quantum communication,” *Sci. Rep.*, vol. 6, no. 1, p. 20463, Feb. 2016, doi: 10.1038/srep20463.

[68] M. Zwerger, A. Pirker, V. Dunjko, H. J. Briegel, and W. Dür, “Long-Range Big Quantum-Data Transmission,” *Phys. Rev. Lett.*, vol. 120, no. 3, p. 030503, Jan. 2018, doi: 10.1103/PhysRevLett.120.030503.

[69] S. Muralidharan, J. Kim, N. Lütkenhaus, M. D. Lukin, and L. Jiang, “Ultrafast and Fault-Tolerant Quantum Communication across Long Distances,” *Phys. Rev. Lett.*, vol. 112, no. 25, p. 250501, Jun. 2014, doi: 10.1103/PhysRevLett.112.250501.

[70] E. Shchukin and P. van Loock, “Optimal Entanglement Swapping in Quantum Repeaters,” *Phys. Rev. Lett.*, vol. 128, no. 15, p. 150502, Apr. 2022, doi: 10.1103/PhysRevLett.128.150502.

[71] M. Mastriani, “Simplified entanglement swapping protocol for the quantum Internet,” *Sci. Rep.*, vol. 13, no. 1, p. 21998, Dec. 2023, doi: 10.1038/s41598-023-49326-4.

[72] E. Shchukin, F. Schmidt, and P. van Loock, “Waiting time in quantum repeaters with probabilistic entanglement swapping,” *Phys. Rev. A*, vol. 100, no. 3, p. 032322, Sep. 2019, doi: 10.1103/PhysRevA.100.032322.

[73] Y. Cao, Y. Zhao, Q. Wang, J. Zhang, S. X. Ng, and L. Hanzo, “The Evolution of Quantum Key Distribution Networks: On the Road to the Qinternet,” *IEEE Commun. Surv. Tutor.*, vol. 24, no. 2, pp. 839–894, 2022, doi: 10.1109/COMST.2022.3144219.

[74] S. Abruzzo, S. Bratzik, N. K. Bernardes, H. Kampermann, P. van Loock, and D. Bruß, “Quantum repeaters and quantum key distribution: Analysis of secret-key rates,” *Phys. Rev. A*, vol. 87, no. 5, p. 052315, May 2013, doi: 10.1103/PhysRevA.87.052315.

[75] Y. Cao, Y. Zhao, J. Li, R. Lin, J. Zhang, and J. Chen, “Hybrid Trusted/Untrusted Relay-Based Quantum Key Distribution Over Optical Backbone Networks,” *IEEE J. Sel. Areas Commun.*, vol. 39, no. 9, pp. 2701–2718, Sep. 2021, doi: 10.1109/JSAC.2021.3064662.

[76] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, “Surface codes: Towards practical large-scale quantum computation,” *Phys. Rev. A*, vol. 86, no. 3, p. 032324, Sep. 2012, doi: 10.1103/PhysRevA.86.032324.

[77] G. D. Forney, “Concatenated codes”.

[78] P.-F. Sun *et al.*, “Deterministic Time-Bin Entanglement between a Single Photon and an Atomic Ensemble,” *Phys. Rev. Lett.*, vol. 128, no. 6, p. 060502, Feb. 2022, doi: 10.1103/PhysRevLett.128.060502.

[79] L. Childress and R. Hanson, “Diamond NV centers for quantum computing and quantum networks,” *MRS Bull.*, vol. 38, no. 2, pp. 134–138, Feb. 2013, doi: 10.1557/mrs.2013.20.

[80] M. Riebe *et al.*, “Deterministic entanglement swapping with an ion-trap quantum computer,” *Nat. Phys.*, vol. 4, no. 11, pp. 839–842, Nov. 2008, doi: 10.1038/nphys1107.

[81] S. Haldar *et al.*, “Reducing classical communication costs in multiplexed quantum repeaters using hardware-aware quasi-local policies,” *Commun. Phys.*, vol. 8, no. 1, pp. 1–20, Apr. 2025, doi: 10.1038/s42005-025-02029-w.

[82] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocol for wireless microsensor networks,” in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, Jan. 2000, p. 10 pp. vol.2-. doi: 10.1109/HICSS.2000.926982.

[83] K. Akkaya and M. Younis, “A survey on routing protocols for wireless sensor networks,” *Ad Hoc Netw.*, vol. 3, no. 3, pp. 325–349, May 2005, doi: 10.1016/j.adhoc.2003.09.010.

[84] J. N. Al-Karaki and A. E. Kamal, “Routing techniques in wireless sensor networks: a survey,” *IEEE Wirel. Commun.*, vol. 11, no. 6, pp. 6–28, Dec. 2004, doi: 10.1109/MWC.2004.1368893.

[85] A. I. Lvovsky, B. C. Sanders, and W. Tittel, “Optical quantum memory,” *Nat. Photonics*, vol. 3, no. 12, pp. 706–714, Dec. 2009, doi: 10.1038/nphoton.2009.231.

[86] K. Heshami *et al.*, “Quantum memories: emerging applications and recent advances,” *J. Mod. Opt.*, vol. 63, no. 20, pp. 2005–2028, Nov. 2016, doi: 10.1080/09500340.2016.1148212.

[87] C. H. Bennett, H. J. Bernstein, S. Popescu, and B. Schumacher, “Concentrating partial entanglement by local operations,” *Phys. Rev. A*, vol. 53, no. 4, pp. 2046–2052, Apr. 1996, doi: 10.1103/PhysRevA.53.2046.

[88] C. H. Bennett, G. Brassard, S. Popescu, B. Schumacher, J. A. Smolin, and W. K. Wootters, “Purification of Noisy Entanglement and Faithful Teleportation via Noisy Channels,” *Phys. Rev. Lett.*, vol. 76, no. 5, pp. 722–725, Jan. 1996, doi: 10.1103/PhysRevLett.76.722.

[89] W. Dür, H.-J. Briegel, J. I. Cirac, and P. Zoller, “Quantum repeaters based on entanglement purification,” *Phys. Rev. A*, vol. 59, no. 1, pp. 169–181, Jan. 1999, doi: 10.1103/PhysRevA.59.169.

[90] M. Mohseni *et al.*, "How to Build a Quantum Supercomputer: Scaling from Hundreds to Millions of Qubits," Jan. 31, 2025, *arXiv*: arXiv:2411.10406. doi: 10.48550/arXiv.2411.10406.

[91] C. Zhou *et al.*, "Realizing all-to-all couplings among detachable quantum modules using a microwave quantum state router," *Npj Quantum Inf.*, vol. 9, no. 1, pp. 1–9, Jun. 2023, doi: 10.1038/s41534-023-00723-7.

[92] X.-Q. Li and Y. Arakawa, "Single qubit from two coupled quantum dots: An approach to semiconductor quantum computations," *Phys. Rev. A*, vol. 63, no. 1, p. 012302, Dec. 2000, doi: 10.1103/PhysRevA.63.012302.

[93] S. B. Rached, C. G. Almudever, E. Alarcon, and S. Abadal, "Spatio-Temporal Characterization of Qubit Routing in Connectivity-Constrained Quantum Processors," Feb. 01, 2024, *arXiv*: arXiv:2402.00469. doi: 10.48550/arXiv.2402.00469.

[94] "Processor types," IBM Quantum Documentation. Accessed: Apr. 04, 2025. [Online]. Available: https://docs.quantum.ibm.com/guides/docs.quantum.ibm.com/guides/processor-types

[95] "SwapGate," IBM Quantum Documentation. Accessed: Apr. 04, 2025. [Online]. Available: https://docs.quantum.ibm.com/api/qiskit/docs.quantum.ibm.com/api/qiskit/qiskit.circuit.library.swapgate

[96] P. Zhao *et al.*, "Quantum Crosstalk Analysis for Simultaneous Gate Operations on Superconducting Qubits," *PRX Quantum*, vol. 3, no. 2, p. 020301, Apr. 2022, doi: 10.1103/PRXQuantum.3.020301.

[97] Z. Zhou, R. Sitler, Y. Oda, K. Schultz, and G. Quiroz, "Quantum Crosstalk Robust Quantum Control," *Phys. Rev. Lett.*, vol. 131, no. 21, p. 210802, Nov. 2023, doi: 10.1103/PhysRevLett.131.210802.

[98] M. Sarovar, T. Proctor, K. Rudinger, K. Young, E. Nielsen, and R. Blume-Kohout, "Detecting crosstalk errors in quantum information processors," *Quantum*, vol. 4, p. 321, Sep. 2020, doi: 10.22331/q-2020-09-11-321.

[99] A. Ovide *et al.*, "Mapping quantum algorithms to multi-core quantum computing architectures," 2023, doi: 10.1109/ISCAS46773.2023.10181589.

[100] "Revisiting the Mapping of Quantum Circuits: Entering the Multi-core Era | ACM Transactions on Quantum Computing." Accessed: Apr. 04, 2025. [Online]. Available: https://dl.acm.org/doi/10.1145/3655029?utm_source=chatgpt.com

[101] H. Shapourian *et al.*, "Quantum Data Center Infrastructures: A Scalable Architectural Design Perspective," Jan. 09, 2025, *arXiv*: arXiv:2501.05598. doi: 10.48550/arXiv.2501.05598.

[102] H. Zhang, K. Yin, A. Wu, H. Shapourian, A. Shabani, and Y. Ding, "MECH: Multi-Entry Communication Highway for Superconducting Quantum Chiplets," in *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2*, in ASPLOS '24, vol. 2. New York, NY, USA: Association for Computing Machinery, Apr. 2024, pp. 699–714. doi: 10.1145/3620665.3640377.

[103] R. V. Meter and S. J. Devitt, "Local and Distributed Quantum Computation," *Computer*, vol. 49, no. 9, pp. 31–42, Sep. 2016, doi: 10.1109/MC.2016.291.

[104] D. Coppersmith, “An approximate Fourier transform useful in quantum factoring,” Jan. 16, 2002, *arXiv*: arXiv:quant-ph/0201067. doi: 10.48550/arXiv.quant-ph/0201067.

[105] N. M. P. Neumann, R. van Houte, and T. Attema, “Imperfect Distributed Quantum Phase Estimation,” in *Computational Science – ICCS 2020*, V. V. Krzhizhanovskaya, G. Závodszky, M. H. Lees, J. J. Dongarra, P. M. A. Sloot, S. Brissos, and J. Teixeira, Eds., Cham: Springer International Publishing, 2020, pp. 605–615. doi: 10.1007/978-3-030-50433-5_46.

[106] S. DiAdamo, M. Ghibaudi, and J. Cruise, “Distributed quantum computing and network control for accelerated vqe,” *IEEE Trans. Quantum Eng.*, vol. 2, pp. 1–21, 2021.

[107] A. Buluç, H. Meyerhenke, I. Safro, P. Sanders, and C. Schulz, “Recent Advances in Graph Partitioning,” in *Algorithm Engineering: Selected Results and Surveys*, L. Kliemann and P. Sanders, Eds., Cham: Springer International Publishing, 2016, pp. 117–158. doi: 10.1007/978-3-319-49487-6_4.

[108] E. Kaur, H. Shapourian, J. Zhao, M. Kilzer, R. Kompella, and R. Nejabati, “Optimized Quantum Circuit Partitioning Across Multiple Quantum Processors,” Jan. 24, 2025, *arXiv*: arXiv:2501.14947. doi: 10.48550/arXiv.2501.14947.

[109] Y. Mao, Y. Liu, and Y. Yang, “Qubit Allocation for Distributed Quantum Computing,” in *IEEE INFOCOM 2023 - IEEE Conference on Computer Communications*, May 2023, pp. 1–10. doi: 10.1109/INFOCOM53939.2023.10228915.

[110] C. Zhang, A. B. Hayes, L. Qiu, Y. Jin, Y. Chen, and E. Z. Zhang, “Time-optimal Qubit mapping,” in *Proceedings of the 26th ACM International Conference on Architectural Support for Programming Languages and Operating Systems*, in ASPLOS ’21. New York, NY, USA: Association for Computing Machinery, Apr. 2021, pp. 360–374. doi: 10.1145/3445814.3446706.

[111] G. Li, Y. Ding, and Y. Xie, “Tackling the Qubit Mapping Problem for NISQ-Era Quantum Devices,” in *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems*, in ASPLOS ’19. New York, NY, USA: Association for Computing Machinery, Apr. 2019, pp. 1001–1014. doi: 10.1145/3297858.3304023.

[112] R. Van Meter, K. Nemoto, W. J. Munro, and K. M. Itoh, “Distributed Arithmetic on a Quantum Multicomputer,” *SIGARCH Comput Arch. News*, vol. 34, no. 2, pp. 354–365, May 2006, doi: 10.1145/1150019.1136517.

[113] D. Ferrari, S. Carretta, and M. Amoretti, “A Modular Quantum Compilation Framework for Distributed Quantum Computing,” *IEEE Trans. Quantum Eng.*, vol. 4, pp. 1–13, 2023, doi: 10.1109/TQE.2023.3303935.

[114] J. Peckham, D. Makaroff, and S. Rayan, “Asynchronous Telegate and Teledata Protocols for Distributed Quantum Computing,” in *2024 IEEE 10th World Forum on Internet of Things (WF-IoT)*, Nov. 2024, pp. 1–6. doi: 10.1109/WF-IoT62078.2024.10811197.

[115] S. Shi and C. Qian, “Concurrent entanglement routing for quantum networks: model and designs,” in Proceedings of the ACM Special Interest Group on Data Communication (SIGCOMM) 2020. New York, NY, USA: Association for Computing Machinery, Jul. 2020, pp. 62–75. doi: 10.1145/3387514.3405853.

[116] M. Pant *et al.*, "Routing entanglement in the quantum internet," *Npj Quantum Inf.*, vol. 5, no. 1, pp. 1–9, Mar. 2019, doi: 10.1038/s41534-019-0139-x.

[117] Z. Yang, A. Ghubaish, R. Jain, H. Shapourian, and A. Shabani, "Asynchronous entanglement routing for the quantum internet," *AVS Quantum Sci.*, vol. 6, no. 1, p. 013801, Jan. 2024, doi: 10.1116/5.0172819.

[118] D.-M. Chiu and R. Jain, "Analysis of the increase and decrease algorithms for congestion avoidance in computer networks," *Comput. Netw. ISDN Syst.*, vol. 17, no. 1, pp. 1–14, Jun. 1989, doi: 10.1016/0169-7552(89)90019-6.

[119] G. Vardoyan and S. Wehner, "Quantum Network Utility Maximization," in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, Sep. 2023, pp. 1238–1248. doi: 10.1109/QCE57702.2023.00140.

[120] S. Pouryousef, H. Shapourian, A. Shabani, and D. Towsley, "Quantum Network Planning for Utility Maximization," in *Proceedings of the 1st Workshop on Quantum Networks and Distributed Quantum Computing*, in QuNet '23. New York, NY, USA: Association for Computing Machinery, Sep. 2023, pp. 13–18. doi: 10.1145/3610251.3610553.

[121] M. Pompili *et al.*, "Experimental demonstration of entanglement delivery using a quantum network stack," *Npj Quantum Inf.*, vol. 8, no. 1, pp. 1–10, Oct. 2022, doi: 10.1038/s41534-022-00631-2.

[122] F. Chen *et al.*, "NISQ Quantum Computing: A Security-Centric Tutorial and Survey [Feature]," *IEEE Circuits Syst. Mag.*, vol. 24, no. 1, pp. 14–32, 2024, doi: 10.1109/MCAS.2024.3349665.

[123] G. Brassard, N. Lütkenhaus, T. Mor, and B. C. Sanders, "Limitations on Practical Quantum Cryptography," *Phys. Rev. Lett.*, vol. 85, no. 6, pp. 1330–1333, Aug. 2000, doi: 10.1103/PhysRevLett.85.1330.

[124] N. Gisin, S. Fasel, B. Kraus, H. Zbinden, and G. Ribordy, "Trojan-horse attacks on quantum-key-distribution systems," *Phys. Rev. A*, vol. 73, no. 2, p. 022320, Feb. 2006, doi: 10.1103/PhysRevA.73.022320.

[125] W. Y. Hwang, "Quantum Key Distribution with High Loss: Toward Global Secure Communication," *Phys. Rev. Lett.*, vol. 91, no. 5, p. 057901, Aug. 2003, doi: 10.1103/PhysRevLett.91.057901.

[126] V. Zapatero *et al.*, "Advances in device-independent quantum key distribution," *Npj Quantum Inf.*, vol. 9, no. 1, pp. 1–11, Feb. 2023, doi: 10.1038/s41534-023-00684-x.

[127] R.-T. Shan, X. Chen, and K.-G. Yuan, "Multi-party blind quantum computation protocol with mutual authentication in network," *Sci. China Inf. Sci.*, vol. 64, no. 6, p. 162302, Apr. 2021, doi: 10.1007/s11432-020-2977-x.

[128] Z. Yang, H. Alfauri, F. Berooh, R. Jain, R. D. Pietro, and A. Erbad, "A survey and comparison of post-quantum and quantum blockchains," *Revis. Submitt. IEEE Commun. Surv. Tutor.*.

[129] Z. Yang *et al.*, "Layer-Wise Security Framework and Analysis for the Quantum Internet," *IEEE J. Sel. Areas Commun.*, Jan. 2025, doi: 10.48550/arXiv.2501.06989.

[130] M. Kalinin and V. Krundyshev, “Security intrusion detection using quantum machine learning techniques,” *J. Comput. Virol. Hacking Tech.*, vol. 19, no. 1, pp. 125–136, Mar. 2023, doi: 10.1007/s11416-022-00435-0.

[131] E. D. Payares and J. C. Martinez-Santos, “Quantum machine learning for intrusion detection of distributed denial of service attacks: a comparative overview,” in *Quantum Computing, Communication, and Simulation*, SPIE, Mar. 2021, pp. 35–43. doi: 10.1117/12.2593297.

[132] J. Li *et al.*, “Fidelity-Guaranteed Entanglement Routing in Quantum Networks,” *IEEE Trans. Commun.*, vol. 70, no. 10, pp. 6748–6763, Oct. 2022, doi: 10.1109/TCOMM.2022.3200115.

[133] L. Yang, Y. Zhao, H. Xu, L. Huang, and C. Qiao, “Dynamic Entanglement Routing Based on Stream Processing for Quantum Networks,” *IEEEACM Trans. Netw.*, pp. 1–16, 2024, doi: 10.1109/TNET.2024.3485919.

[134] Z. Yang, A. Ghubaish, R. Jain, R. Kompella, and H. Shapourian, “Multi-Tree Quantum Routing in Realistic Topologies,” *IEEE Commun. Mag.*, vol. 62, no. 10, pp. 78–85, Oct. 2024, doi: 10.1109/MCOM.006.2300851.

[135] R. Negrin, N. Dirnegger, W. Munizzi, J. Talukdar, and P. Narang, “Efficient Multiparty Entanglement Distribution with DODAG-X Protocol,” Aug. 13, 2024, *arXiv*: arXiv:2408.07118. doi: 10.48550/arXiv.2408.07118.

[136] E. Schoute, L. Mancinska, T. Islam, I. Kerenidis, and S. Wehner, “Shortcuts to quantum network routing,” Oct. 17, 2016, *arXiv*: arXiv:1610.05238. doi: 10.48550/arXiv.1610.05238.

[137] S. Bahrani, R. Wang, J. Parra-Ullauri, R. D. Oliveira, R. Nejabati, and D. Simeonidou, “Joint Network and Computing Resource Optimisation in Distributed Quantum Computing,” in *2024 Optical Fiber Communications Conference and Exhibition (OFC)*, Mar. 2024, pp. 1–3. Accessed: Dec. 26, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10526825

[138] S. Bahrani, R. D. Oliveira, J. M. Parra-Ullauri, R. Wang, and D. Simeonidou, “Resource Management and Circuit Scheduling for Distributed Quantum Computing Interconnect Networks,” Oct. 14, 2024, *arXiv*: arXiv:2409.12675. doi: 10.48550/arXiv.2409.12675.

[139] S. Pouryousef *et al.*, “Resource Management in Quantum Virtual Private Networks,” in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, Sep. 2023, pp. 393–394. doi: 10.1109/QCE57702.2023.10298.

[140] S. Pouryousef, H. Shapourian, A. Shabani, R. Kompella, and D. Towsley, “Resource Placement for Rate and Fidelity Maximization in Quantum Networks,” *IEEE Trans. Quantum Eng.*, vol. 5, pp. 1–16, 2024, doi: 10.1109/TQE.2024.3432390.

[141] L. Le, T. N. Nguyen, A. Lee, and B. Dumba, “Entanglement Routing For Quantum Networks: A Deep Reinforcement Learning Approach,” in *ICC 2022 - IEEE International Conference on Communications*, May 2022, pp. 395–400. doi: 10.1109/ICC45855.2022.9839240.

[142] P. Sharma, S. Gupta, V. Bhatia, and S. Prakash, “Deep reinforcement learning-based routing and resource assignment in quantum key distribution-secured optical networks,” *IET Quantum Commun.*, vol. 4, no. 3, pp. 136–145, 2023, doi: 10.1049/qtc2.12063.

[143] J. I. Cirac, A. K. Ekert, S. F. Huelga, and C. Macchiavello, "Distributed quantum computation over noisy channels," *Phys. Rev. A*, vol. 59, no. 6, pp. 4249–4254, Jun. 1999, doi: 10.1103/PhysRevA.59.4249.

[144] A. Y. Kitaev, "Quantum computations: algorithms and error correction," *Russ. Math. Surv.*, vol. 52, no. 6, p. 1191, Dec. 1997, doi: 10.1070/RM1997v052n06ABEH002155.

[145] D. Ferrari, A. S. Cacciapuoti, M. Amoretti, and M. Caleffi, "Compiler Design for Distributed Quantum Computing," *IEEE Trans. Quantum Eng.*, vol. 2, pp. 1–20, 2021, doi: 10.1109/TQE.2021.3053921.

[146] N. M. P. Neumann and R. S. Wezeman, "Distributed Quantum Machine Learning," vol. 1585, 2022, pp. 281–293. doi: 10.1007/978-3-031-06668-9_20.

[147] H. K. Lo, M. Curty, and B. Qi, "Measurement-Device-Independent Quantum Key Distribution," *Phys. Rev. Lett.*, vol. 108, no. 13, p. 130503, Mar. 2012, doi: 10.1103/PhysRevLett.108.130503.

[148] P. Jouguet, S. Kunz-Jacques, A. Leverrier, P. Grangier, and E. Diamanti, "Experimental demonstration of long-distance continuous-variable quantum key distribution," *Nat. Photonics*, vol. 7, no. 5, pp. 378–381, May 2013, doi: 10.1038/nphoton.2013.63.

[149] L. Lydersen, C. Wiechers, C. Wittmann, D. Elser, J. Skaar, and V. Makarov, "Thermal blinding of gated detectors in quantum cryptography," *Opt. Express*, vol. 18, no. 26, pp. 27938–27954, Dec. 2010, doi: 10.1364/OE.18.027938.

[150] P. Lodahl *et al.*, "Chiral quantum optics," *Nature*, vol. 541, no. 7638, pp. 473–480, Jan. 2017, doi: 10.1038/nature21037.

[151] B. Heim *et al.*, "Atmospheric channel characteristics for quantum communication with continuous polarization variables," *Appl. Phys. B*, vol. 98, no. 4, pp. 635–640, Mar. 2010, doi: 10.1007/s00340-009-3838-8.

[152] S. J. van Enk, N. Lütkenhaus, and H. J. Kimble, "Experimental procedures for entanglement verification," *Phys. Rev. A*, vol. 75, no. 5, p. 052318, May 2007, doi: 10.1103/PhysRevA.75.052318.

[153] H. Lee, J. Lim, and H. Yang, "Quantum direct communication with authentication," *Phys. Rev. A*, vol. 73, no. 4, p. 042305, Apr. 2006, doi: 10.1103/PhysRevA.73.042305.

[154] D. Joseph *et al.*, "Transitioning organizations to post-quantum cryptography," *Nature*, vol. 605, no. 7909, pp. 237–243, May 2022, doi: 10.1038/s41586-022-04623-2.

[155] C. Tian, Z. Yang, and R. Jain, "Asynchronous Entanglement Routing for Simultaneous User in Quantum Networks," *submitted*, 2025.

[156] C. Tian, Z. Yang, and R. Jain, "Asynchronous Routing for Multipartite Entanglement in Quantum Networks," *submitted*, 2025.

[157] Z. Yang *et al.*, "Resource Management for Distributed Quantum Computing in Quantum Data Centers," *submitted*, 2025.

[158] Z. Yang, T. Salman, R. Jain, and R. D. Pietro, "Decentralization Using Quantum Blockchain: A Theoretical Analysis," *IEEE Trans. Quantum Eng.*, vol. 3, pp. 1–16, 2022, doi: 10.1109/TQE.2022.3207111.

[159] H.-K. Lo, M. Curty, and K. Tamaki, "Secure quantum key distribution," *Nat. Photonics*, vol. 8, no. 8, Art. no. 8, Aug. 2014, doi: 10.1038/nphoton.2014.149.

[160] P. Kómár *et al.*, "A quantum network of clocks," *Nat. Phys.*, vol. 10, no. 8, pp. 582–587, Aug. 2014, doi: 10.1038/nphys3000.

[161] R. Beals *et al.*, "Efficient distributed quantum computing," *Proc. R. Soc. Math. Phys. Eng. Sci.*, vol. 469, no. 2153, p. 20120686, May 2013, doi: 10.1098/rspa.2012.0686.

[162] D. A. Lidar and T. A. Brun, *Quantum error correction*. Cambridge University Press, 2013.

[163] J.-W. Pan, D. Bouwmeester, H. Weinfurter, and A. Zeilinger, "Experimental entanglement swapping: entangling photons that never interacted," *Phys. Rev. Lett.*, vol. 80, no. 18, pp. 3891–3894, May 1998, doi: 10.1103/PhysRevLett.80.3891.

[164] B. M. Terhal, "Quantum error correction for quantum memories," *Rev. Mod. Phys.*, vol. 87, no. 2, pp. 307–346, Apr. 2015, doi: 10.1103/RevModPhys.87.307.

[165] A. Tchebotareva *et al.*, "Entanglement between a diamond spin qubit and a photonic time-bin qubit at telecom wavelength," *Phys. Rev. Lett.*, vol. 123, no. 6, p. 063601, Aug. 2019, doi: 10.1103/PhysRevLett.123.063601.

[166] A. Dréau, A. Tchebotareva, A. E. Mahdaoui, C. Bonato, and R. Hanson, "Quantum frequency conversion of single photons from a Nitrogen-Vacancy center in diamond to telecommunication wavelengths," *Phys. Rev. Appl.*, vol. 9, no. 6, p. 064031, Jun. 2018, doi: 10.1103/PhysRevApplied.9.064031.

[167] S. Wengerowsky *et al.*, "Passively stable distribution of polarisation entanglement over 192 km of deployed optical fibre," *Npj Quantum Inf.*, vol. 6, no. 1, pp. 1–5, Jan. 2020, doi: 10.1038/s41534-019-0238-8.

[168] A. Acín, J. I. Cirac, and M. Lewenstein, "Entanglement percolation in quantum networks," *Nat. Phys.*, vol. 3, no. 4, pp. 256–259, Apr. 2007, doi: 10.1038/nphys549.

[169] S. Perseguers, M. Lewenstein, A. Acín, and J. I. Cirac, "Quantum random networks," *Nat. Phys.*, vol. 6, no. 7, pp. 539–543, Jul. 2010, doi: 10.1038/nphys1665.

[170] C. Li, T. Li, Y. X. Liu, and P. Cappellaro, "Effective routing design for remote entanglement generation on quantum networks," *Npj Quantum Inf.*, vol. 7, no. 1, pp. 1–12, Jan. 2021, doi: 10.1038/s41534-020-00344-4.

[171] A. Patil, M. Pant, D. Englund, D. Towsley, and S. Guha, "Entanglement generation in a quantum network at distance-independent rate," *Npj Quantum Inf.*, vol. 8, no. 1, pp. 1–9, May 2022, doi: 10.1038/s41534-022-00536-0.

[172] M. Caleffi, "Optimal routing for quantum networks," *IEEE Access*, vol. 5, pp. 22299–22312, 2017, doi: 10.1109/ACCESS.2017.2763325.

[173] S. DiAdamo, B. Qi, G. Miller, R. Kompella, and A. Shabani, "Packet switching in quantum networks: A path to the quantum Internet," *Phys. Rev. Res.*, vol. 4, no. 4, p. 043064, Oct. 2022, doi: 10.1103/PhysRevResearch.4.043064.

[174] D. Xu, Y. Chen, Y. Xiong, C. Qiao, and X. He, "On the complexity of and algorithms for finding the shortest path with a disjoint counterpart," *IEEEACM Trans. Netw.*, vol. 14, no. 1, pp. 147–158, Feb. 2006, doi: 10.1109/TNET.2005.863451.

[175] K. Chakraborty, F. Rozpedek, A. Dahlberg, and S. Wehner, "Distributed routing in a quantum internet," Jul. 26, 2019, *arXiv*: arXiv:1907.11630. doi: 10.48550/arXiv.1907.11630.

[176] R. Alexander *et al.*, "RPL: IPv6 routing protocol for low-power and lossy networks," Internet Engineering Task Force, Request for Comments RFC 6550, Mar. 2012. Accessed: Feb. 20, 2023. [Online]. Available: https://datatracker.ietf.org/doc/rfc6550

[177] R. G. Gallager, P. A. Humblet, and P. M. Spira, "A distributed algorithm for minimum-weight spanning trees," *ACM Trans. Program. Lang. Syst. TOPLAS*, vol. 5, no. 1, pp. 66–77, 1983.

[178] E. Korach, S. Moran, and S. Zaks, "Tight lower and upper bounds for some distributed algorithms for a complete network of processors," in *Proceedings of the third annual ACM symposium on Principles of distributed computing*, in PODC '84. New York, NY, USA: Association for Computing Machinery, Aug. 1984, pp. 199–207. doi: 10.1145/800222.806747.

[179] O. Malik, X. Meng, S. Havlin, G. Korniss, B. K. Szymanski, and J. Gao, "Concurrence percolation threshold of large-scale quantum networks," *Commun. Phys.*, vol. 5, no. 1, pp. 1–11, Jul. 2022, doi: 10.1038/s42005-022-00958-4.

[180] S. R. Broadbent and J. M. Hammersley, "Percolation processes: I. Crystals and mazes," *Math. Proc. Camb. Philos. Soc.*, vol. 53, no. 3, pp. 629–641, Jul. 1957, doi: 10.1017/S0305004100032680.

[181] H. Kesten, "The critical probability of bond percolation on the square lattice equals 1/2," *Commun. Math. Phys.*, vol. 74, no. 1, pp. 41–59, Feb. 1980, doi: 10.1007/BF01197577.

[182] A. Hagberg, D. Schult, and P. Swart, "NetworkX reference," *Release 30*, 2023, [Online]. Available: https://networkx.org/documentation/stable/_downloads/networkx_reference.pdf

[183] G. W. Furnas and J. Zacks, "Multitrees: enriching and reusing hierarchical structure," presented at the the SIGCHI conference, Boston, Massachusetts, United States: ACM Press, 1994, pp. 330–336. doi: 10.1145/191666.191778.

[184] P. Erdős and A. Rényi, "On random graphs," *Publ. Math. Debr.*, vol. 6, no. 290–297, p. 18, 1959.

[185] "About ESnet," ESnet. Accessed: Nov. 15, 2023. [Online]. Available: https://www.es.net/about/

[186] S. Knight, H. X. Nguyen, N. Falkner, R. Bowden, and M. Roughan, "The Internet Topology Zoo," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 9, pp. 1765–1775, Oct. 2011, doi: 10.1109/JSAC.2011.111002.

[187] "About Internet2," Internet2. Accessed: Nov. 15, 2023. [Online]. Available: https://internet2.edu/community/about-us/

[188] R. Nejabati *et al.*, "Quantum Data Center Interconnect - A Cisco Vision," Cisco Research. Accessed: Dec. 21, 2024. [Online]. Available: https://research-strapi-s3.s3.us-east-2.amazonaws.com/QN_4_QDC_2024_Release3_2_7c0273901f.pdf

[189] IBM, "First IBM Quantum Data Center in Europe Opens; Will Include IBM's Most Performant Quantum Systems," IBM Newsroom. Accessed: Dec. 21, 2024. [Online].

Available: https://newsroom.ibm.com/2024-10-01-first-ibm-quantum-data-center-in-europe-opens-will-include-ibms-most-performant-quantum-systems

[190] D. Cuomo, M. Caleffi, and A. S. Cacciapuoti, "Towards a distributed quantum computing ecosystem," *IET Quantum Commun.*, vol. 1, no. 1, pp. 3–8, 2020, doi: 10.1049/iet-qtc.2020.0002.

[191] M. Caleffi, A. S. Cacciapuoti, and G. Bianchi, "Quantum internet: from communication to distributed computing!," in *Proceedings of the 5th ACM International Conference on Nanoscale Computing and Communication*, in NANOCOM '18. New York, NY, USA: Association for Computing Machinery, Sep. 2018, pp. 1–4. doi: 10.1145/3233188.3233224.

[192] P. Andrés-Martínez and C. Heunen, "Automated distribution of quantum circuits via hypergraph partitioning," *Phys. Rev. A*, vol. 100, no. 3, p. 032308, Sep. 2019, doi: 10.1103/PhysRevA.100.032308.

[193] R. G Sundaram, H. Gupta, and C. R. Ramakrishnan, "Efficient Distribution of Quantum Circuits," in *35th International Symposium on Distributed Computing (DISC 2021)*, S. Gilbert, Ed., in Leibniz International Proceedings in Informatics (LIPIcs), vol. 209. Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021, p. 41:1-41:20. doi: 10.4230/LIPIcs.DISC.2021.41.

[194] A. Singla, P. B. Godfrey, and A. Kolla, "High Throughput Data Center Topology Design," presented at the 11th USENIX Symposium on Networked Systems Design and Implementation (NSDI 14), 2014, pp. 29–41. Accessed: Dec. 26, 2024. [Online]. Available: https://www.usenix.org/conference/nsdi14/technical-sessions/presentation/singla

[195] B. Lebiednik, A. Mangal, and N. Tiwari, "A Survey and Evaluation of Data Center Network Topologies," May 05, 2016, *arXiv*: arXiv:1605.01701. doi: 10.48550/arXiv.1605.01701.

[196] Y. Mao, Y. Liu, X. Shang, and Y. Yang, "Network Topology Design for Distributed Quantum Computing," in *2024 IEEE 44th International Conference on Distributed Computing Systems (ICDCS)*, Jul. 2024, pp. 1213–1223. doi: 10.1109/ICDCS60910.2024.00115.

[197] R. Yu, R. Dutta, and J. Liu, "On Topology Design for the Quantum Internet," *IEEE Netw.*, vol. 36, no. 5, pp. 64–70, Sep. 2022, doi: 10.1109/MNET.001.2200170.

[198] G. Vardoyan, S. Guha, P. Nain, and D. Towsley, "On the Stochastic Analysis of a Quantum Entanglement Switch," *SIGMETRICS Perform Eval Rev*, vol. 47, no. 2, pp. 27–29, Dec. 2019, doi: 10.1145/3374888.3374899.

[199] M. A. Hall, J. B. Altepeter, and P. Kumar, "Ultrafast Switching of Photonic Entanglement," *Phys. Rev. Lett.*, vol. 106, no. 5, p. 053901, Feb. 2011, doi: 10.1103/PhysRevLett.106.053901.

[200] C. Clos, "A Study of Non-Blocking Switching Networks," *Bell Syst. Tech. J.*, vol. 32, no. 2, pp. 406–424, 1953, doi: 10.1002/j.1538-7305.1953.tb01433.x.

[201] K. Bilal *et al.*, "Quantitative comparisons of the state-of-the-art data center architectures," *Concurr. Comput. Pract. Exp.*, vol. 25, no. 12, pp. 1771–1783, 2013, doi: 10.1002/cpe.2963.

[202] O. Milul *et al.*, "Superconducting Cavity Qubit with Tens of Milliseconds Single-Photon Coherence Time," *PRX Quantum*, vol. 4, no. 3, p. 030336, Sep. 2023, doi: 10.1103/PRXQuantum.4.030336.

[203] M. Cococcioni and L. Fiaschi, "The Big-M method with the numerical infinite M," *Optim. Lett.*, vol. 15, no. 7, pp. 2455–2468, Oct. 2021, doi: 10.1007/s11590-020-01644-6.

[204] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms, fourth edition*. MIT Press, 2022.

[205] IBM, "OPL and MILP - IBM ILOG CPLEX Optimization Studio." Accessed: Dec. 29, 2024. [Online]. Available: https://www.ibm.com/docs/en/icos/22.1.1?topic=problem-opl-milp

[206] "NetworkX — NetworkX documentation." Accessed: Dec. 30, 2024. [Online]. Available: https://networkx.org/

[207] J. Y. Yen, "An algorithm for finding shortest routes from all source nodes to a given destination in general networks," *Q. Appl. Math.*, vol. 27, no. 4, pp. 526–530, 1970, doi: 10.1090/qam/253822.

[208] R. K. Jain, D.-M. W. Chiu, and W. R. Hawe, "A quantitative measure of fairness and discrimination," *East. Res. Lab. Digit. Equip. Corp. Hudson MA*, vol. 21, 1984.

[209] C.-Y. Lu, Y. Cao, C.-Z. Peng, and J.-W. Pan, "Micius quantum experiments in space," *Rev. Mod. Phys.*, vol. 94, no. 3, p. 035001, Jul. 2022, doi: 10.1103/RevModPhys.94.035001.

[210] Z. Yang, M. Zolanvari, and R. Jain, "A survey of important issues in quantum computing and communications," *IEEE Commun. Surv. Tutor.*, vol. 25, no. 2, pp. 1059–1094, 2023, doi: 10.1109/COMST.2023.3254481.

[211] A. S. Cacciapuoti, M. Caleffi, F. Tafuri, F. S. Cataliotti, S. Gherardini, and G. Bianchi, "Quantum internet: Networking challenges in distributed quantum computing," *IEEE Netw.*, vol. 34, no. 1, pp. 137–143, 2019.

[212] Y. Zhao, C. H. F. Fung, B. Qi, C. Chen, and H. K. Lo, "Quantum hacking: Experimental demonstration of time-shift attack against practical quantum-key-distribution systems," *Phys. Rev. A*, vol. 78, no. 4, p. 042333, Oct. 2008, doi: 10.1103/PhysRevA.78.042333.

[213] A. B. Price, J. G. Rarity, and C. Erven, "A quantum key distribution protocol for rapid denial of service detection," *EPJ Quantum Technol.*, vol. 7, no. 1, Art. no. 1, Dec. 2020, doi: 10.1140/epjqt/s40507-020-00084-6.

[214] M. Manzo, T. Roosta, and S. Sastry, "Time synchronization attacks in sensor networks," in *Proceedings of the 3rd ACM Workshop on Security of Ad hoc and Sensor Networks*, in (SASN) 2005. New York, NY, USA: Association for Computing Machinery, Nov. 2005, pp. 107–116. doi: 10.1145/1102219.1102238.

[215] V. Scarani, H. Bechmann-Pasquinucci, N. J. Cerf, M. Dušek, N. Lütkenhaus, and M. Peev, "The security of practical quantum key distribution," *Rev. Mod. Phys.*, vol. 81, no. 3, pp. 1301–1350, Sep. 2009, doi: 10.1103/RevModPhys.81.1301.

[216] Y. Nagamatsu, A. Mizutani, R. Ikuta, T. Yamamoto, N. Imoto, and K. Tamaki, "Security of quantum key distribution with light sources that are not independently and identically distributed," *Phys. Rev. A*, vol. 93, no. 4, p. 042325, Apr. 2016, doi: 10.1103/PhysRevA.93.042325.

[217] X. Ma, B. Qi, Y. Zhao, and H. K. Lo, "Practical decoy state for quantum key distribution," *Phys. Rev. A*, vol. 72, no. 1, p. 012326, Jul. 2005, doi: 10.1103/PhysRevA.72.012326.

[218] C. Santori, D. Fattal, and Y. Yamamoto, *Single-photon Devices and Applications*. John Wiley & Sons, 2010.

[219] J. Y. Hu *et al.*, "Experimental quantum secure direct communication with single photons," *Light Sci. Appl.*, vol. 5, no. 9, pp. e16144–e16144, Sep. 2016, doi: 10.1038/lsa.2016.144.

[220] N. Jain, E. Anisimova, I. Khan, V. Makarov, C. Marquardt, and G. Leuchs, "Trojan-horse attacks threaten the security of practical quantum cryptography," *New J. Phys.*, vol. 16, no. 12, p. 123030, Dec. 2014, doi: 10.1088/1367-2630/16/12/123030.

[221] F. G. Deng, X. H. Li, H. Y. Zhou, and Z. jun Zhang, "Improving the security of multiparty quantum secret sharing against Trojan horse attack," *Phys. Rev. A*, vol. 72, no. 4, p. 044302, Oct. 2005, doi: 10.1103/PhysRevA.72.044302.

[222] N. Jain, B. Stiller, I. Khan, V. Makarov, C. Marquardt, and G. Leuchs, "Risk Analysis of Trojan-Horse Attacks on Practical Quantum Key Distribution Systems," *IEEE J. Sel. Top. Quantum Electron.*, vol. 21, no. 3, pp. 168–177, May 2015, doi: 10.1109/JSTQE.2014.2365585.

[223] D. Gottesman, H. K. Lo, N. Lutkenhaus, and J. Preskill, "Security of quantum key distribution with imperfect devices," in *International Symposium on Information Theory (ISIT) 2004. Proceedings.*, Jun. 2004, p. 136. doi: 10.1109/ISIT.2004.1365172.

[224] L. Lydersen, C. Wiechers, C. Wittmann, D. Elser, J. Skaar, and V. Makarov, "Hacking commercial quantum cryptography systems by tailored bright illumination," *Nat. Photonics*, vol. 4, no. 10, Art. no. 10, Oct. 2010, doi: 10.1038/nphoton.2010.214.

[225] L. Lydersen, M. K. Akhlaghi, A. H. Majedi, J. Skaar, and V. Makarov, "Controlling a superconducting nanowire single-photon detector using tailored bright illumination," *New J. Phys.*, vol. 13, no. 11, p. 113042, Nov. 2011, doi: 10.1088/1367-2630/13/11/113042.

[226] F. Xu, B. Qi, and H. K. Lo, "Experimental demonstration of phase-remapping attack in a practical quantum key distribution system," *New J. Phys.*, vol. 12, no. 11, p. 113026, Nov. 2010, doi: 10.1088/1367-2630/12/11/113026.

[227] B. P. Williams, K. A. Britt, and T. S. Humble, "Tamper-Indicating Quantum Seal," *Phys. Rev. Appl.*, vol. 5, no. 1, p. 014001, Jan. 2016, doi: 10.1103/PhysRevApplied.5.014001.

[228] M. Peev *et al.*, "The SECOQC quantum key distribution network in Vienna," *New J. Phys.*, vol. 11, no. 7, p. 075001, Jul. 2009, doi: 10.1088/1367-2630/11/7/075001.

[229] H. E. Brandt, "Quantum-cryptographic entangling probe," *Phys. Rev. A*, vol. 71, no. 4, p. 042312, Apr. 2005, doi: 10.1103/PhysRevA.71.042312.

[230] H. Azuma, "An entangling-probe attack on Shor's algorithm for factorization," *J. Mod. Opt.*, vol. 65, no. 4, pp. 415–422, Feb. 2018, doi: 10.1080/09500340.2017.1397221.

[231] D. S. Naik, C. G. Peterson, A. G. White, A. J. Berglund, and P. G. Kwiat, "Entangled State Quantum Cryptography: Eavesdropping on the Ekert Protocol," *Phys. Rev. Lett.*, vol. 84, no. 20, pp. 4733–4736, May 2000, doi: 10.1103/PhysRevLett.84.4733.

[232] R. L. Rivest and A. Shamir, "How to expose an eavesdropper," *Commun. ACM*, vol. 27, no. 4, pp. 393–394, Apr. 1984, doi: 10.1145/358027.358053.

[233] K. Svozil, “Feasibility of the interlock protocol against man-in-the-middle attacks on quantum cryptography,” *Int. J. Quantum Inf.*, vol. 03, no. 04, pp. 649–654, Dec. 2005, doi: 10.1142/S0219749905001511.

[234] M. S. Byrd and D. A. Lidar, “Combined error correction techniques for quantum computing architectures,” *J. Mod. Opt.*, vol. 50, no. 8, pp. 1285–1297, May 2003, doi: 10.1080/09500340308235203.

[235] M. Mehic *et al.*, “A Novel Approach to Quality-of-Service Provisioning in Trusted Relay Quantum Key Distribution Networks,” *IEEE ACM Trans. Netw.*, vol. 28, no. 1, pp. 168–181, Feb. 2020, doi: 10.1109/TNET.2019.2956079.

[236] E. G. Abdallah, H. S. Hassanein, and M. Zulkernine, “A Survey of Security Attacks in Information-Centric Networking,” *IEEE Commun. Surv. Tutor.*, vol. 17, no. 3, pp. 1441–1454, 2015, doi: 10.1109/COMST.2015.2392629.

[237] K. a. G. Fisher *et al.*, “Quantum computing on encrypted data,” *Nat. Commun.*, vol. 5, no. 1, p. 3074, Jan. 2014, doi: 10.1038/ncomms4074.

[238] P. Hilaire, E. Barnes, S. E. Economou, and F. Grosshans, “Error-correcting entanglement swapping using a practical logical photon encoding,” *Phys. Rev. A*, vol. 104, no. 5, p. 052623, Nov. 2021, doi: 10.1103/PhysRevA.104.052623.

[239] A. Dahlberg *et al.*, “A link layer protocol for quantum networks,” in *Proceedings of the ACM Special Interest Group on Data Communication (SIGCOMM)*, Beijing China: ACM, Aug. 2019, pp. 159–173. doi: 10.1145/3341302.3342070.

[240] R. Van Meter and J. Touch, “Designing quantum repeater networks,” *IEEE Commun. Mag.*, vol. 51, no. 8, pp. 64–71, 2013.

[241] R. Van Meter *et al.*, “A Quantum Internet Architecture,” in *2022 IEEE International Conference on Quantum Computing and Engineering (QCE)*, Sep. 2022, pp. 341–352. doi: 10.1109/QCE53715.2022.00055.

[242] T. Mihara, “Quantum identification schemes with entanglements,” *Phys. Rev. A*, vol. 65, no. 5, p. 052326, May 2002, doi: 10.1103/PhysRevA.65.052326.

[243] M. Dušek, O. Haderka, M. Hendrych, and R. Myška, “Quantum identification system,” *Phys. Rev. A*, vol. 60, no. 1, pp. 149–156, Jul. 1999, doi: 10.1103/PhysRevA.60.149.

[244] “Graph generators — NetworkX 3.1 documentation.” Accessed: Sep. 22, 2023. [Online]. Available: https://networkx.org/documentation/stable/reference/generators.html

[245] C. Cicconetti, M. Conti, and A. Passarella, “Quality of Service in Quantum Networks,” *IEEE Netw.*, vol. 36, no. 5, pp. 24–31, 2022.

[246] Z. Li, K. Xue, J. Li, N. Yu, D. S. L. Wei, and R. Li, “Connection-Oriented and Connectionless Remote Entanglement Distribution Strategies in Quantum Networks,” *IEEE Netw.*, vol. 36, no. 6, pp. 150–156, Nov. 2022, doi: 10.1109/MNET.107.2100483.

[247] V. Ramachandran and S. Nandi, “Bleeding Edge DDoS Mitigation Techniques for ISPs,” in *8th International Conference on Information Technology*,

[248] S. T. Zargar, J. Joshi, and D. Tipper, “A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks,” *IEEE Commun. Surv. Tutor.*, vol. 15, no. 4, pp. 2046–2069, 2013.

[249] S. Ranjan, R. Swaminathan, M. Uysal, A. Nucci, and E. Knightly, "DDoS-shield: DDoS-resilient scheduling to counter application layer attacks," *IEEE ACM Trans. Netw.*, vol. 17, no. 1, pp. 26–39, 2008.

[250] I. Aad, J. P. Hubaux, and E. W. Knightly, "Denial of service resilience in ad hoc networks," in *Proceedings of the 10th annual international conference on Mobile computing and networking*, Philadelphia PA USA: ACM, Sep. 2004, pp. 202–215. doi: 10.1145/1023720.1023741.

[251] J. Wang, N. Juarez, E. Kohm, Y. Liu, J. Yuan, and H. Song, "Integration of SDR and UAS for malicious Wi-Fi hotspots detection," in *2019 Integrated Communications, Navigation and Surveillance Conference (ICNS)*, IEEE, 2019, pp. 1–8.

[252] L. Kamin, E. Shchukin, F. Schmidt, and P. Van Loock, "Exact rate analysis for quantum repeaters with imperfect memories and entanglement swapping as soon as possible," *Phys. Rev. Res.*, vol. 5, no. 2, p. 023086, May 2023, doi: 10.1103/PhysRevResearch.5.023086.

[253] L. Gyongyosi and S. Imre, "Decentralized base-graph routing for the quantum internet," *Phys. Rev. A*, vol. 98, no. 2, p. 022310, Aug. 2018, doi: 10.1103/PhysRevA.98.022310.

[254] A. Yimsiriwattana and S. J. L. Jr, "Distributed quantum computing: a distributed Shor algorithm," in *Quantum Information and Computation II*, SPIE, Aug. 2004, pp. 360–372. doi: 10.1117/12.546504.

[255] A. Belapurkar, A. Chakrabarti, H. Ponnapalli, N. Varadarajan, S. Padmanabhuni, and S. Sundarrajan, *Distributed Systems Security: Issues, Processes and Solutions*. John Wiley & Sons, 2009.

[256] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proceedings of the Twenty-eighth Annual ACM Symposium on Theory of Computing (STOC) 1996*, Philadelphia, Pennsylvania, United States: ACM Press, 1996, pp. 212–219. doi: 10.1145/237814.237866.

[257] D. Qiu, L. Luo, and L. Xiao, "Distributed Grover's algorithm," *Theor. Comput. Sci.*, vol. 993, p. 114461, Apr. 2024, doi: 10.1016/j.tcs.2024.114461.

[258] P. W. Shor, "Fault-tolerant quantum computation," in *Proceedings of 37th conference on foundations of computer science*, IEEE, 1996, pp. 56–65.

[259] E. R. Bennewitz, F. Hopfmueller, B. Kulchytskyy, J. Carrasquilla, and P. Ronagh, "Neural Error Mitigation of Near-Term Quantum Simulations," *Nat. Mach. Intell.*, vol. 4, no. 7, pp. 618–624, Jul. 2022, doi: 10.1038/s42256-022-00509-0.

[260] T. Kim, I. Stork Genannt Wersborg, F. N. C. Wong, and J. H. Shapiro, "Complete physical simulation of the entangling-probe attack on the Bennett-Brassard 1984 protocol," *Phys. Rev. A*, vol. 75, no. 4, p. 042327, Apr. 2007, doi: 10.1103/PhysRevA.75.042327.

[261] R. Garcia-Patron and N. J. Cerf, "Unconditional optimality of Gaussian attacks against continuous-variable QKD," *Phys. Rev. Lett.*, vol. 97, no. 19, p. 190503, Nov. 2006, doi: 10.1103/PhysRevLett.97.190503.

[262] H. Inamori, L. Rallan, and V. Vedral, "Security of EPR-based quantum cryptography against incoherent symmetric attacks," *J. Phys. Math. Gen.*, vol. 34, no. 35, p. 6913, 2001.

[263] M. Li *et al.*, "Measurement-device-independent quantum key distribution with modified coherent state," *Opt. Lett.*, vol. 39, no. 4, pp. 880–883, 2014.

[264] E. Biham and T. Mor, "Security of Quantum Cryptography against Collective Attacks," *Phys. Rev. Lett.*, vol. 78, no. 11, pp. 2256–2259, Mar. 1997, doi: 10.1103/PhysRevLett.78.2256.

[265] T. Salman, M. Zolanvari, A. Erbad, R. Jain, and M. Samaka, "Security Services Using Blockchains: A State of the Art Survey," *IEEE Commun. Surv. Tutor.*, vol. 21, no. 1, pp. 858–880, 2019, doi: 10.1109/COMST.2018.2863956.

[266] M. Nofer, P. Gomber, O. Hinz, and D. Schiereck, "Blockchain," *Bus. Inf. Syst. Eng. Int. J. Wirtsch.*, vol. 59, no. 3, pp. 183–187, 2017.

[267] T. Renduchintala, H. Alfauri, Z. Yang, R. D. Pietro, and R. Jain, "A Survey of Blockchain Applications in the FinTech Sector," *J. Open Innov. Technol. Mark. Complex.*, vol. 8, no. 4, Art. no. 4, Dec. 2022, doi: 10.3390/joitmc8040185.

[268] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," Bitcoin. Accessed: Jan. 29, 2023. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[269] P. W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *Proceedings 35th Annual Symposium on Foundations of Computer Science*, Nov. 1994, pp. 124–134. doi: 10.1109/SFCS.1994.365700.

[270] A. Hosoyamada and Y. Sasaki, "Quantum Collision Attacks on Reduced SHA-256 and SHA-512," 2021, 2021/292. Accessed: Apr. 15, 2025. [Online]. Available: https://eprint.iacr.org/2021/292

[271] S. S. Tannu and M. K. Qureshi, "Not All Qubits Are Created Equal: A Case for Variability-Aware Policies for NISQ-Era Quantum Computers," in *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems*, in ASPLOS '19. New York, NY, USA: Association for Computing Machinery, Apr. 2019, pp. 987–999. doi: 10.1145/3297858.3304007.

[272] B. Rodenburg and S. P. Pappas, "Blockchain and quantum computing," *MITRE Tech. Rep.*, 2017, Accessed: Apr. 15, 2025. [Online]. Available: https://apps.dtic.mil/sti/trecms/pdf/AD1125436.pdf

[273] J. J. Kearney and C. A. Perez-Delgado, "Vulnerability of blockchain technologies to quantum attacks," *Array*, vol. 10, p. 100065, Jul. 2021, doi: 10.1016/j.array.2021.100065.

[274] D. J. Bernstein and T. Lange, "Post-quantum cryptography," *Nature*, vol. 549, no. 7671, Art. no. 7671, Sep. 2017, doi: 10.1038/nature23461.

[275] T. M. Fernández-Caramès and P. Fraga-Lamas, "Towards Post-Quantum Blockchain: A Review on Blockchain Cryptography Resistant to Quantum Computing Attacks," *IEEE Access*, vol. 8, pp. 21091–21116, 2020, doi: 10.1109/ACCESS.2020.2968985.

[276] D. Rajan and M. Visser, "Quantum Blockchain Using Entanglement in Time," *Quantum Rep.*, vol. 1, no. 1, Art. no. 1, Sep. 2019, doi: 10.3390/quantum1010002.

[277] C. Simon, "Towards a global quantum network," *Nat. Photonics*, vol. 11, no. 11, Art. no. 11, Nov. 2017, doi: 10.1038/s41566-017-0032-0.

[278] M. Edwards, A. Mashatan, and S. Ghose, “A review of quantum and hybrid quantum/classical blockchain protocols,” *Quantum Inf. Process.*, vol. 19, no. 6, p. 184, May 2020, doi: 10.1007/s11128-020-02672-y.

[279] E. O. Kiktenko *et al.*, “Quantum-secured blockchain,” *Quantum Sci. Technol.*, vol. 3, no. 3, p. 035004, May 2018, doi: 10.1088/2058-9565/aabc6b.

[280] X. Sun, Q. Wang, P. Kulicki, and X. Zhao, “Quantum-enhanced Logic-based Blockchain I: Quantum Honest-success Byzantine Agreement and Qulogicoin,” Jul. 16, 2018, *arXiv*: arXiv:1805.06768. doi: 10.48550/arXiv.1805.06768.

[281] Y. Liu, D. He, M. S. Obaidat, N. Kumar, M. K. Khan, and K.-K. Raymond Choo, “Blockchain-based identity management systems: A review,” *J. Netw. Comput. Appl.*, vol. 166, p. 102731, Sep. 2020, doi: 10.1016/j.jnca.2020.102731.

[282] “Ethereum Whitepaper,” ethereum.org. Accessed: Apr. 15, 2025. [Online]. Available: https://ethereum.org/en/whitepaper/

[283] Y.-L. Gao, X.-B. Chen, Y.-L. Chen, Y. Sun, X.-X. Niu, and Y.-X. Yang, “A Secure Cryptocurrency Scheme Based on Post-Quantum Blockchain,” *IEEE Access*, vol. 6, pp. 27205–27213, 2018, doi: 10.1109/ACCESS.2018.2827203.

[284] F. Ablayev, D. Bulychkov, D. Sapaev, A. Vasiliev, and M. Ziatdinov, “Quantum-Assisted Blockchain,” *Lobachevskii J. Math.*, Accessed: Apr. 15, 2025. [Online]. Available: https://link.springer.com/article/10.1134/S1995080218070028

[285] X. Zhang, “One-way quantum identity authentication based on public key,” *Chin. Sci. Bull.*, vol. 54, no. 12, pp. 2018–2021, Jun. 2009, doi: 10.1007/s11434-009-0350-9.

[286] X. Sun, M. Sopek, Q. Wang, and P. Kulicki, “Towards Quantum-Secured Permissioned Blockchain: Signature, Consensus, and Logic,” *Entropy*, vol. 21, no. 9, Art. no. 9, Sep. 2019, doi: 10.3390/e21090887.

[287] D. Shrier, W. Wu, and A. Pentland, “Blockchain & infrastructure (identity, data security),” *Mass. Inst. Technol.-Connect. Sci.*, vol. 1, no. 3, pp. 1–19, 2016.

[288] “Blockchain & Cryptocurrency News, Investing Guides, Market Forecasts & Reviews,” Bitnation. Accessed: Apr. 15, 2025. [Online]. Available: https://bitnation.co/

[289] “Consensys - A complete suite of trusted products to build anything in web3,” Consensys. Accessed: Apr. 15, 2025. [Online]. Available: https://consensys.io

[290] “Stacks - The Leading Bitcoin L2 for Smart Contracts, Apps, DeFi.” Accessed: Apr. 15, 2025. [Online]. Available: https://www.stacks.co/

[291] “PingOne Neo.” Accessed: Apr. 15, 2025. [Online]. Available: https://www.pingidentity.com/en/lp/ac/pingone-neo.html

[292] X. Zhu and Y. Badr, “A Survey on Blockchain-Based Identity Management Systems for the Internet of Things,” in *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, Jul. 2018, pp. 1568–1573. doi: 10.1109/Cybermatics_2018.2018.00263.

[293] G. Iovane, “Computational quantum key distribution (CQKD) on decentralized ledger and blockchain,” *J. Discrete Math. Sci. Cryptogr.*, vol. 24, no. 4, pp. 1021–1042, May 2021, doi: 10.1080/09720529.2020.1820691.

[294] K. Ikeda, “qBitcoin: A Peer-to-Peer Quantum Cash System,” in *Intelligent Computing*, K. Arai, S. Kapoor, and R. Bhatia, Eds., in Advances in Intelligent Systems and Computing. Cham: Springer International Publishing, 2019, pp. 763–771. doi: 10.1007/978-3-030-01174-1_58.

[295] Y.-L. Gao, X.-B. Chen, G. Xu, K.-G. Yuan, W. Liu, and Y.-X. Yang, “A novel quantum blockchain scheme base on quantum entanglement and DPoS,” *Quantum Inf. Process.*, vol. 19, no. 12, p. 420, Nov. 2020, doi: 10.1007/s11128-020-02915-y.

[296] C. Li, Y. Xu, J. Tang, and W. Liu, “Quantum Blockchain: A Decentralized, Encrypted and Distributed Database Based on Quantum Mechanics,” *J. Quantum Comput.*, vol. 1, no. 2, pp. 49–63, 2019, doi: 10.32604/jqc.2019.06715.

[297] H. Buhrman, R. Cleve, J. Watrous, and R. de Wolf, “Quantum Fingerprinting,” *Phys. Rev. Lett.*, vol. 87, no. 16, p. 167902, Sep. 2001, doi: 10.1103/PhysRevLett.87.167902.

[298] C. ho Hong, J. Heo, J. G. Jang, and D. Kwon, “Quantum identity authentication with single photon,” *Quantum Inf. Process.*, vol. 16, no. 10, p. 236, Aug. 2017, doi: 10.1007/s11128-017-1681-0.

[299] A. Kawachi and C. Portmann, “On the Power of Quantum Encryption Keys,” in *Post-Quantum Cryptography*, J. Buchmann and J. Ding, Eds., Berlin, Heidelberg: Springer, 2008, pp. 165–180. doi: 10.1007/978-3-540-88403-3_12.

[300] A. Ambainis, M. Mosca, A. Tapp, and R. De Wolf, “Private quantum channels,” in *Proceedings 41st Annual Symposium on Foundations of Computer Science*, Nov. 2000, pp. 547–553. doi: 10.1109/SFCS.2000.892142.

[301] M. Liang and L. Yang, “Public-key encryption and authentication of quantum information,” *Sci. China Phys. Mech. Astron.*, vol. 55, no. 9, pp. 1618–1629, Sep. 2012, doi: 10.1007/s11433-011-4806-y.

[302] L. Yang, M. Liang, B. Li, L. Hu, and D.-G. Feng, “Quantum public-key cryptosystems based on induced trapdoor one-way transformations,” Jul. 12, 2011, *arXiv*: arXiv:1012.5249. doi: 10.48550/arXiv.1012.5249.

[303] C. Sabín, B. Peropadre, M. del Rey, and E. Martín-Martínez, “Extracting Past-Future Vacuum Correlations Using Circuit QED,” *Phys. Rev. Lett.*, vol. 109, no. 3, p. 033602, Jul. 2012, doi: 10.1103/PhysRevLett.109.033602.

[304] D. M. Greenberger, M. A. Horne, and A. Zeilinger, “Going beyond Bell’s theorem,” in *Bell’s Theorem, Quantum Theory and Conceptions of the Universe*, M. Kafatos, Ed., in Fundamental Theories of Physics. , Dordrecht: Springer Netherlands, 1989, pp. 69–72. Accessed: Feb. 20, 2023. [Online]. Available: https://doi.org/10.1007/978-94-017-0849-4_10

[305] G. Carvacho, F. Graffitti, V. D’Ambrosio, B. C. Hiesmayr, and F. Sciarrino, “Experimental investigation on the geometry of GHZ states,” *Sci. Rep.*, vol. 7, no. 1, p. 13265, Oct. 2017, doi: 10.1038/s41598-017-13124-6.

[306] E. Megidish, A. Halevy, T. Shacham, T. Dvir, L. Dovrat, and H. S. Eisenberg, "Entanglement Swapping between Photons that have Never Coexisted," *Phys. Rev. Lett.*, vol. 110, no. 21, p. 210403, May 2013, doi: 10.1103/PhysRevLett.110.210403.

[307] D. Edward Bruschi, C. Sabín, A. White, V. Baccetti, D. K. L. Oi, and I. Fuentes, "Testing the effects of gravity and motion on quantum entanglement in space-based experiments," *New J. Phys.*, vol. 16, no. 5, p. 053041, May 2014, doi: 10.1088/1367-2630/16/5/053041.

[308] S. Banerjee, A. Mukherjee, and P. K. Panigrahi, "Quantum blockchain using weighted hypergraph states," *Phys. Rev. Res.*, vol. 2, no. 1, p. 013322, Mar. 2020, doi: 10.1103/PhysRevResearch.2.013322.

[309] A. J. Bennet and S. Daryanoosh, "Energy efficient mining on a quantum-enabled blockchain using light," *Ledger*, vol. 4, Jul. 2019, doi: 10.5195/ledger.2019.143.

[310] W. McCutcheon *et al.*, "Experimental verification of multipartite entanglement in quantum networks," *Nat. Commun.*, vol. 7, no. 1, Art. no. 1, Nov. 2016, doi: 10.1038/ncomms13251.

[311] A. Mostefaoui and M. Raynal, "Leader-based consensus," *Parallel Process. Lett.*, vol. 11, no. 01, pp. 95–107, Mar. 2001, doi: 10.1142/S0129626401000452.

[312] K. Pinter, D. Schmelz, R. Lamber, S. Strobl, and T. Grechenig, "Towards a Multi-party, Blockchain-Based Identity Verification Solution to Implement Clear Name Laws for Online Media Platforms," in *Business Process Management: Blockchain and Central and Eastern Europe Forum*, C. Di Ciccio, R. Gabryelczyk, L. García-Bañuelos, T. Hernaus, R. Hull, M. Indihar Štemberger, A. Kő, and M. Staples, Eds., Cham: Springer International Publishing, 2019, pp. 151–165. doi: 10.1007/978-3-030-30429-4_11.

[313] M. Al-Bassam, "SCPKI: A Smart Contract-based PKI and Identity System," in *Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts*, in BCC '17. New York, NY, USA: Association for Computing Machinery, Apr. 2017, pp. 35–40. doi: 10.1145/3055518.3055530.

[314] J. A. Cortese and T. M. Braje, "Loading Classical Data into a Quantum Computer," Mar. 05, 2018, *arXiv*: arXiv:1803.01958. doi: 10.48550/arXiv.1803.01958.

[315] J. Yard, P. Hayden, and I. Devetak, "Quantum Broadcast Channels," *IEEE Trans. Inf. Theory*, vol. 57, no. 10, pp. 7147–7162, Oct. 2011, doi: 10.1109/TIT.2011.2165811.

[316] S. Bäuml and K. Azuma, "Fundamental limitation on quantum broadcast networks," *Quantum Sci. Technol.*, vol. 2, no. 2, p. 024004, May 2017, doi: 10.1088/2058-9565/aa6d3c.

[317] C. Brukner, S. Taylor, S. Cheung, and V. Vedral, "Quantum Entanglement in Time," Feb. 18, 2004, *arXiv*: arXiv:quant-ph/0402127. doi: 10.48550/arXiv.quant-ph/0402127.

[318] K. Mitarai, M. Kitagawa, and K. Fujii, "Quantum analog-digital conversion," *Phys. Rev. A*, vol. 99, no. 1, p. 012301, Jan. 2019, doi: 10.1103/PhysRevA.99.012301.

[319] O. Dib, K.-L. Brousmiche, A. Durand, E. Thea, and E. B. Hamida, "Consortium blockchains: Overview, applications and challenges," *Int J Adv Telecommun*, vol. 11, no. 1, pp. 51–64, 2018.

[320] M. Zolanvari, Z. Yang, K. Khan, R. Jain, and N. Meskin, "TRUST XAI: Model-Agnostic Explanations for AI With a Case Study on IIoT Security," *IEEE Internet Things J.*, vol. 10, no. 4, pp. 2967–2978, Feb. 2023, doi: 10.1109/JIOT.2021.3122019.

Appendix A. Quantum Network Stack

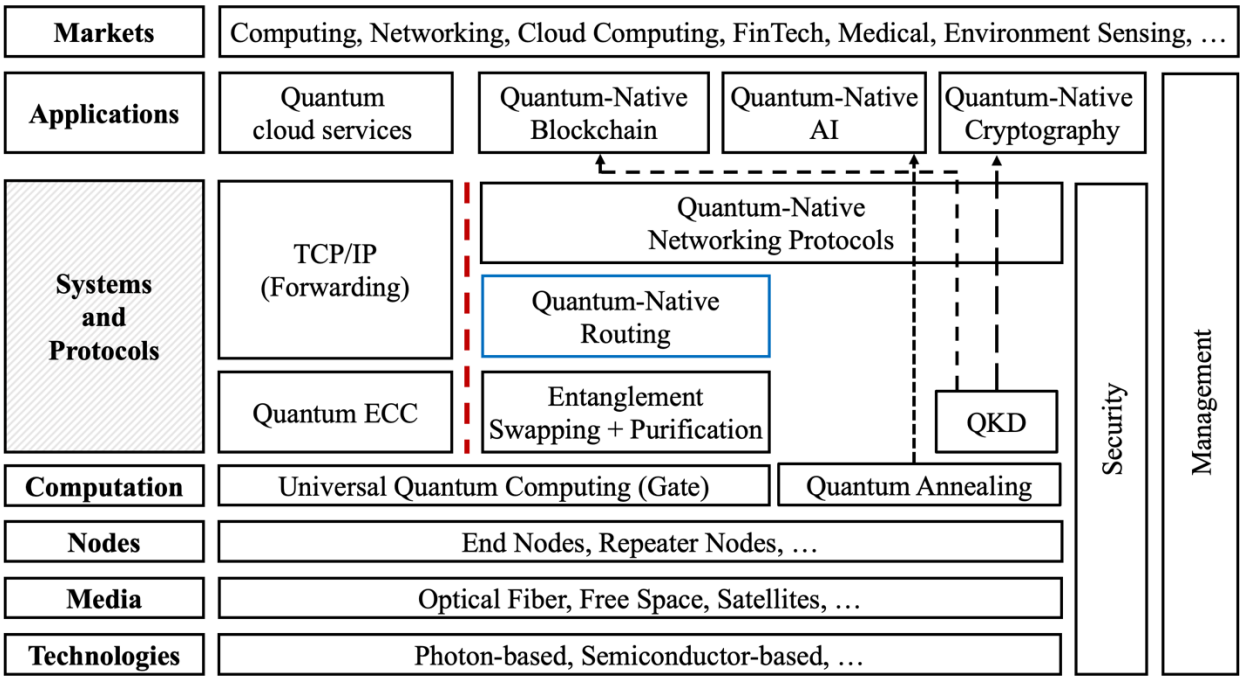


Figure A.1. *7-layer Stack of Quantum Research.*

We have identified seven layers within the technical framework for quantum research, providing a schematic to grasp the diverse elements essential to quantum technology, as illustrated in Figure A.1. At the foundational level, the stack is anchored by diverse technologies, which include photon-based and semiconductor-based approaches [1], critical for quantum states and operations. These technologies interface with various media, such as optical fiber and free space, such as satellites, providing the physical medium for quantum communication and interactions. Nodes in the quantum network, which encompass end nodes and repeater Nodes, form the fundamental infrastructure, facilitating quantum computation and communication processes. In the computation realm, universal quantum computing (i.e., gate model) and Quantum Annealing are the primary mechanisms that execute quantum algorithms and solve complex problems. Systems and protocols

are the backbone of any quantum network. While TCP/IP represents conventional networking standards, the stack introduces quantum-native protocols with unique functionalities like entanglement swapping and asynchronous routing. At the Applications layer, solutions such as quantum cloud services, quantum blockchains, quantum artificial intelligence, and quantum cryptography showcase the potential real-world uses of quantum technologies [128], [158], [159], [320], [267]. Lastly, at the markets layer, the stack emphasizes the wide-ranging sectors that quantum technologies can influence, e.g., computing, networking, clouds, FinTech, medical, and sensing, underlining the vast impact quantum advancements can have across industries.

Our observation shows that while substantial progress has been made in the lower layers, the upper layers have received comparatively less attention. These layers reflect research initiatives in physics and electrical engineering [4], [15]. In contrast, the upper layers indicate endeavors in software, encompassing network protocols and applications. In particular, within the layer of systems and protocols, there are two distinct approaches: one involves using quantum repeaters that transmit quantum information (left of the red dashed line). The transmission of quantum information through repeaters requires error correction using a significant number of error-correcting qubits, which is restricted by the current capabilities of quantum computers. The other involves distributing entanglement as a communication resource (i.e., quantum-native routing), which is our primary interest and, in our view, deserves greater attention from the research community.

Appendix B. DODAG Loop Avoidance

When nodes select routes in a graph, loops can cause the routing process not to conclude, wasting time and resources. Therefore, they must be detected and avoided. While a directed acyclic graph (DAG) does not form closed loops, a DODAG may do so if a node is greedy and wants to select as many parents as possible. To prevent this, a rank-based non-greedy mechanism is used to detect loops before allowing a node to join the DODAG.

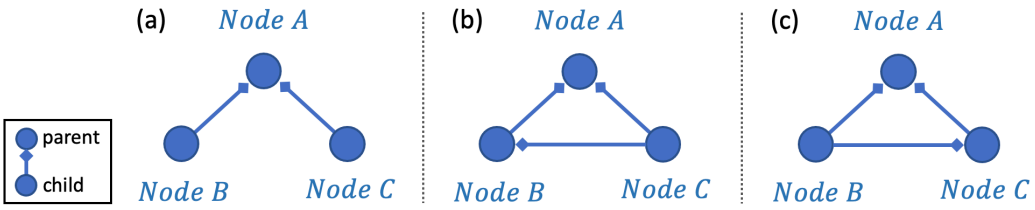


Figure B.1. *DODAG Greedy Parent Selection.*

If a node in the DODAG is greedy and wants to move deeper (closer to the root) by increasing its number of parent nodes, there is a risk of creating a loop. Consider Figure B.1(a), which depicts the initial state of a branch of a DODAG, where node A has two children, B and C. Let us assume that A is the closest node to the root among the three, making it the preferred parent of B and C. If nodes B and C are both eager to select more parents, a loop may occur. For example, node C may want to choose both A and B as parents, making it deeper than both A and B, as shown in Figure B.1(b). Suppose node B is also greedy and decides to select C as a parent against the non-greedy rule. In this case, node B will intentionally leave the DODAG and then rejoin at a lower rank, taking both A and C as parents, as illustrated in Figure B.1(c). As a result, node B will become deeper than both A and C. Node C, who is also greedy, will then leave and rejoin at a lower rank to get two parents and have a lower rank than both of them. Node B will repeat the process by

leaving and rejoining at a lower rank, which will cause the DODAG to alternate between Figure B.1(b) and (c), creating an infinite cycle. This situation can be prevented by ensuring nodes B and C do not select parents from deeper nodes or leave the DODAG intentionally to acquire more parents. Instead, they should maintain a reasonable rank relative to their preferred parent, A. Additionally, they should not process parenting requests from deeper nodes. In the example, as long as node B does not depart from the DODAG solely to acquire two parents, the branch of the DODAG will remain in the state depicted in Figure B.1(b).

Appendix C. GHS Delay Scenarios

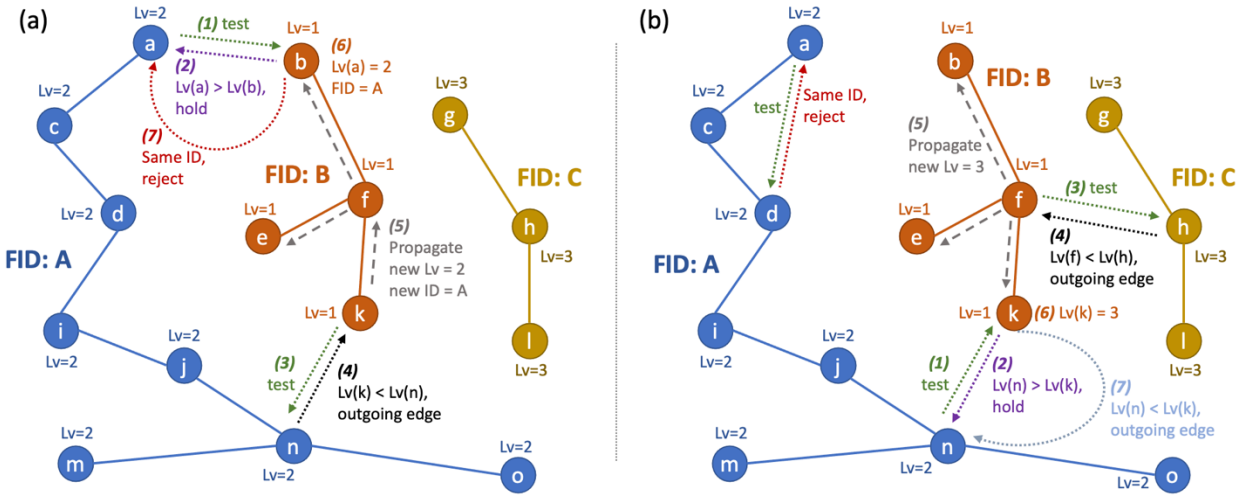


Figure C.1. *Fragment Combination in the GHS Algorithm.*

In Figure C.1, the procedures labeled (1) through (7) operate following the rules outlined in Subsection 2.4.2 Two scenarios are described as follows: C.1(a) The "test" message from Node a to b is rejected due to the outgoing edge determined at (4). However, if no level value is available to determine whether an edge is outgoing, a loop will be formed through nodes "a-b-f-k-n-j-i-d-c-a." In Figure C.1(b), if the level of the node that initiates a "test" message is less than or equal to the level of the recipient node, then it is safe to consider the edge in-between them outgoing. No loop will be formed in this case. An entanglement link can safely be created for this edge.

That is, in Figure C.1(a), (1) Node a sends a "test" message to Node b, but they cannot determine which ID is the latest because $Lv(a) > Lv(b)$. (2) Consequently, they put the edge on hold. (3) Meanwhile, another node, k, initiates a "test" message to Node n. (4) Since $Lv(k) < Lv(n)$, an outgoing edge is established. (5) Node k propagates the new level and ID to other nodes in the same fragment. (6) Node b now knows that its latest level is 2 and its latest ID is A. (7) Nodes a

and *b* also realize they have the same ID and leave the edge alone. In Figure C.1(b), a similar process leads to finding an outgoing edge successfully. These processes use level increases to avoid broadcast delay, which may cause loops or unnecessary connections. Thus, fragment splitting while maintaining the level value will not affect the merge or absorb operations in the GHS algorithm. If a link breaks during broadcasting, the on-hold edge will remain on hold until the adjacent nodes broadcast the breaking message. Then, it will release the on-hold status and start the next "test" message. If edge "k-f" breaks in Figure C.1(a) and (b), only process (2) will remain on hold until the node knows it is isolated. On the other hand, if edge "i-j" breaks, Fragment A splits into two fragments with different IDs, but it will not impact any operations from (1) to (7).

Appendix D. Mean Path Length Approximation

We can examine the physical 2D grid topology by representing it as a 2D square lattice G comprising of n nodes where$\sqrt{n} \in \mathbb{Z}$. We consider the L1 distance between every two nodes within graph G. The L1 distance for nodes s and t with coordinates $s = (x_s, y_s)$ and $t = (x_t, y_t)$ is determined as follows:

$$d(s,t) = d\big((x_s, y_s), (x_t, y_t)\big) = |x_s - x_t| + |y_s - y_t| \qquad \text{(D.1)}$$

The length of the mth shortest self-avoiding path between s and t is as follows (c.f., [179]):

$$l_{s,t}^{m} = d(s,t) + 2(m-1) \qquad \text{(D.2)}$$

Let us denote the total number of self-avoiding paths of length l_m between s and t as $N_{l_m}^{s,t}$. This value can be obtained by the piecewise path enumeration algorithm introduced in Reference [179]. Then, we can represent the mean length of all the possible paths with maximum length l_m between s and t as

$$l_{s,t} = \frac{\sum_{i=1}^{m} l_i N_{l_i}^{s,t}}{\sum_{i=1}^{m} N_{l_i}^{s,t}} \qquad \text{(D.3)}$$

Assuming l_m represents the longest path length between nodes s and t in G, the above expression corresponds to the average length of all possible paths connecting them. Nevertheless, to simplify the computation without compromising the outcome, we can assign a sufficiently large, fixed value to m without necessarily determining the precise average length of all possible paths. Note that the maximum value of m is proportional to n.

Appendix E. Iterative Approximation

Here, we develop an iterative approximation for n, the number of rounds of purification required to achieve a target fidelity f_i, by examining the behavior of the fidelity purification function based on (8), assuming the same initial Bell pair fidelity, represented as follows:

$$F_{n+1} = \frac{{F_n}^2}{{F_n}^2 + (1 - F_n)^2} \qquad \text{(E.1)}$$

(E.1) can be simplified under the assumption that F_n is close to 1, which is usually the case in practical scenarios where high fidelity is needed, and initial fidelities are also reasonably high.

When F_n is close to 1, $(1 - F_n)$ is small. Let us denote $x_n = 1 - F_n$. The update formula then simplifies to:

$$x_{n+1} = 1 - F_{n+1} = 1 - \frac{{F_n}^2}{{F_n}^2 + (1 - F_n)^2} = \frac{(1 - F_n)^2}{{F_n}^2 + (1 - F_n)^2} \approx (1 - F_n)^2 \qquad \text{(E.2)}$$

Thus:

$$x_{n+1} \approx x_n^2 \qquad \text{(E.3)}$$

Given (E.3), we can see that the error rate decreases quadratically with each round. If we start with $x_0 = 1 - F_0$, after n rounds, x_n will be approximately:

$$x_n \approx (1 - F_0)^{2^n} \qquad \text{(E.4)}$$

To achieve a target fidelity f_i, we need:

$$1 - f_i \leq (1 - F_0)^{2^n} \qquad \text{(E.5)}$$

Taking logarithms on both sides, we get:

$$\log(1 - f_i) \leq 2^n \log(1 - F_0) \qquad \text{(E.6)}$$

Thus:

$$n \geq \frac{\log(\log(1 - f_i)/\log(1 - F_0))}{log2} \qquad \text{(E.7)}$$