Harnessing multi-partite entanglement in quantum networks

Distribution strategies and utilization for anonymous communication

vorgelegt von

Jan Scholtens de Jong, M. Sc.
ORCID: 0000-0001-9662-9337

an der Fakultät IV - Elektrotechnik und Informatik
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Naturwissenschaften
- Dr. rer. nat. -

genehmigte Dissertation

Promotionsausschuss:

Vorsitzende:	Prof. Dr. Jean-Pierre Seifert
Gutachter:	Dr. Anna Pappa
Gutachter:	Dr. Damian Markham
Gutachter:	Dr. Tobias Heindel

Tag der wissenschaftlichen Aussprache: 07. März 2025

Berlin 2025

Abstract

The field of quantum communication concerns the distribution of *quantum information* in networks, encoded into quantum states of matter. By leveraging properties of quantum information that are unique to quantum physics, it can develop advanced network communication methods that overcome the limitations of classical physics. *Quantum cryptography*, a closely related field, details the use of quantum information for secure communication, in particular through *quantum key distribution*, which promises unconditionally secure communication.

Entanglement — a quintessential property of quantum states for which there exists no classical counterpart — is a ubiquitous resource fundamental to this field. It manifests as correlations between two or more quantum systems ('*bi-partite*' and '*multi-partite*' entanglement, respectively) that cannot be explained by classical physics.

This thesis presents my two main research contributions. First, it provides a theoretical study of multi-partite entanglement, presenting analytical tools to evaluate and compare its myriad different forms, additionally studying the potential *equivalence* between these forms.

Second, it presents the utilization of multi-partite entanglement in cryptographic tasks, specifically focusing on *anonymous conference key agreement* (ACKA). ACKA protocols allow any number of nodes in a network to secretly communicate, while keeping their identities private.

The research presented in this thesis demonstrates that multi-partite entanglement can enhance communication protocols compared to traditional bi-partite approaches, contributing to the ongoing development of quantum communication technologies and the emerging vision of a global quantum internet.

Zusammenfassung

Das Feld der Quantenkommunikation befasst sich mit der Verteilung von *quantum information* in Netzwerken, kodiert in *quantum states of matter*. Durch die Nutzung von Eigenschaften der *quantum information*, die einzigartig für die Quantenphysik sind, können fortschrittliche Kommunikationsmethoden für Netzwerke entwickelt werden, die die Einschränkungen der klassischen Physik überwinden. Die *quantum cryptography*, ein eng verwandtes Gebiet, behandelt den Einsatz von *quantum information* für sichere Kommunikation, insbesondere durch *quantum key distribution*, die bedingungslos sichere Kommunikation verspricht.

Entanglement — eine grundlegende Eigenschaft von *quantum states*, für die es kein klassisches Pendant gibt — ist eine allgegenwärtige Ressource, die für dieses Feld von zentraler Bedeutung ist. Sie zeigt sich als Korrelation zwischen zwei oder mehr Quantensystemen (*bi-partite* bzw. *multi-partite* Entanglement), die nicht durch klassische Physik erklärbar sind.

Diese Dissertation präsentiert meine zwei Hauptforschungsbeiträge. Erstens wird eine theoretische Untersuchung von *multi-partite entanglement* vorgestellt, die analytische Werkzeuge zur Bewertung und zum Vergleich der zahlreichen unterschiedlichen Formen liefert. Darüber hinaus wird die mögliche *equivalence* zwischen diesen Formen untersucht.

Zweitens wird die Nutzung von *multi-partite entanglement* in kryptographischen Aufgaben präsentiert, mit besonderem Fokus auf *anonymous conference key agreement* (ACKA). ACKA-Protokolle ermöglichen es beliebigen Knoten in einem Netzwerk, geheim zu kommunizieren, während ihre Identitäten privat bleiben.

Die in dieser Dissertation vorgestellte Forschung zeigt, dass *multi-partite entanglement* Kommunikationsprotokolle im Vergleich zu traditionellen *bi-partite* Ansätzen verbessern kann und leistet einen Beitrag zur Weiterentwicklung der Quantenkommunikationstechnologien sowie zur Verwirklichung der aufkommenden Vision eines globalen Quanteninternets.

Dedicated to Jan Scholtens Folkers,
my grandfather and my namesake.
He did what I did, 60 years ago.

Preface

> Many years later, as he faced the thesis committee,
> Colonel Aureliano Buendía was to remember that distant
> afternoon when he decided to do a PhD.
>
> Gabriel García Márquez, *One Hundred Years of Solitude*
> (paraphrased)

This thesis is the culmination of the research that I have performed during the last years, in the context of my PhD. In this research, I have explored different aspects of the exciting field of quantum communication and cryptography. This growing and flourishing field aims to utilize new paradigms on the boundary between physics, mathematics and computer science (and an ever-so-slight touch of philosophy, if you so please), combining all these sciences to develop secure communication, blazing-fast networks, and novel applications that would have been understood as science-fiction five decades ago.

The field of quantum information science saw a tremendous increase in popularity over the last one or two decades, which has been coined the *second quantum revolution*. Many people are hopeful that two will be enough, and that the current endeavours in research and development will be able to carry the field beyond its academic roots, and lift it to be the disruptive, powerful new technology it aspires to be. For quantum communication specifically, this would culminate in the vision of a global *quantum internet*, that allows anyone on earth to participate in quantum communication tasks.

Only time will tell if this will be the case, but one thing is clear: there are many open questions and tasks that have yet to be answered before a full-fledged quantum internet is realised. My research has aimed to answer some of these questions, presenting new protocols, methods and tools to be used in quantum communication.

Jarn (Jan Scholtens) de Jong
Berlin, 2024

Contents

Referencing and notational style xiii

List of figures xviii

List of tables xix

Introduction xxi

Structure of this thesis xxvii

I Mathematical Properties of Quantum Networks

1 Mathematical preliminaries 3

1.1 The Pauli group . . . 4
1.2 Quantum states . . . 7
1.3 Operations on qubits . . . 11
1.4 Entropies . . . 15
1.5 Entangled states . . . 17
1.6 The Clifford group and local unitary operations . . . 20

2 The stabilizer formalism 23

2.1 Stabilizer states . . . 24
2.2 Unitary evolutions of stabilizer states . . . 26
2.3 Measurements on stabilizer states . . . 28
2.4 Reduced states and bipartite entanglement . . . 32
2.5 Conclusion and further reading . . . 35

3 Graph states 37

3.1 Graphs . . . 38
3.2 Graph states . . . 42
3.3 Local complementations and local Clifford operations . . . 46
3.4 Single-qubit Pauli measurements on graph states . . . 49
3.5 Conclusion and further reading . . . 55

II Multi-partite Entanglement in Quantum Networks

4 Local operations on stabilizer states 59
4.1 Local operations and the LOCC paradigm . . . 61
4.2 Reduction to graph states . . . 63
4.3 Orbits and entanglement classes . . . 64
4.4 Local equivalence of graph states . . . 68
4.5 Equivalence involving measurements . . . 71
4.6 Conclusion . . . 72
5 Extracting GHZ states by local operations 75
5.1 Setting . . . 76
5.2 Impossible extraction patterns . . . 77
5.3 The maximal extraction pattern . . . 79
5.4 Reduction of other patterns . . . 82
5.5 Implementations . . . 85
5.6 Conclusion . . . 89
6 Characterizing Entanglement 91
6.1 The reduced stabilizer for a graph state . . . 93
6.2 The rank of reduced states as an invariant . . . 97
6.3 Identifiers derived from the rank invariant . . . 97
6.4 Performance of the identifiers . . . 102
6.5 Different LC-orbits with equal identifiers . . . 108
6.6 Efficiency of the introduced methods . . . 109
6.7 Conclusion . . . 110

III Anonymous Conference Key Agreement

7 Introduction to Quantum Cryptography 115
7.1 Basics of cryptography . . . 116
7.2 Introduction to quantum key distribution . . . 119
7.3 Security of QKD . . . 125
7.4 Generalization to more than two parties . . . 132
7.5 Anonymity in networking protocols . . . 133
7.6 Conclusion . . . 135

8 Star Network ACKA 137
8.1 Setting for the security and the protocols . . . 138
8.2 Original protocol . . . 140
8.3 Improved protocols . . . 148
8.4 Conclusion and discussion on network topology . . . 155
9 Linear Network ACKA 157
9.1 Setting . . . 159
9.2 Protocol statement . . . 160
9.3 Security and anonymity . . . 164
9.4 Performance . . . 166
9.5 Discussion and conclusion . . . 169
10 Experimental Realizations of ACKA 175
10.1 Star network ACKA . . . 176
10.2 Linear network ACKA . . . 178
10.3 Conclusion . . . 183

IV Conclusion

11 Conclusion 187
Future research 189
Acknowledgements 193

V Bibliography

List of publications 197
Bibliography 199

Appendices

A Proof of Thm. 1 218
B Corrections for GHZ extraction in chapter 5 220
C Details for fidelity estimation method of chapter 5 223

D SUBPROTOCOLS OF ACKA 225
D.1 NOTIFICATION . . . 225
D.2 ANONYMOUS MULTIPARTITE ENTANGLEMENT . . . 228
D.3 VERIFICATION . . . 230
E ANONYMITY IN ACKA 233
E.1 Anonymity during AME . . . 234
E.2 Anonymity during VERIFICATION . . . 238
E.3 Anonymity during the KeyGen rounds . . . 241
F CORRECTIONS DURING LINACKA 242
F.1 Detailing the necessary corrections . . . 242
F.2 Calculating the corrections . . . 244
G SECURITY PROOF OF LINACKA 247
G.1 Protocol statement . . . 247
G.2 Security proof . . . 249
H ANONYMITY PROOF OF LINACKA 254
H.1 Proof of anonymity . . . 254

Referencing and notational style

This chapter details and explains style in referencing and in notation, and naming conventions that are used in this thesis.

Referencing

All references are clickable links, and all clickable links are coloured. All the links that refer to content elsewhere (i.e. *citations*) are coloured blue. Any link that refers to content *within* this thesis is coloured red. Depending on what it refers to, different styling is used:

- Equations are not referenced with any preposition, i.e. they are referenced as (1.1). For example, 'Combining (1.1) with (1.6), it follows that...'. An exception is made when they occur at the start of a sentence, then e.g. 'Eq. (1.1) shows that...' will be written.
- Chapter and section references are abbreviated in the middle of a sentence, and not abbreviated at the start of a sentence. For example, 'Chapter 1 introduces the topic of...' and 'In sec. 1.1, the topic of...'.
- References to definitions, theorems and corollaries, are always abbreviated and capitalized. For example, 'Def. 1 gives the...' and 'From Thm. 1, it is...'.
- References to figures and tables are similar, but also in small-caps and boldface, e.g. '...which is shown in **Fig.** 3.1.', or '**Tab.** 1.1 shows...'.
- References to protocols are never abbreviated, and are in monospace. Moreover, they are numbered with roman numerals, e.g. 'The steps of `Protocol I` are....'.
- Citations are always written '[1]', and never e.g. 'Ref. [1]'.
- My own publications are referenced separately using the abbreviation 'Pub.' and are numbered with alphabetical characters, e.g. 'Pub. **[A]** contains the first...', or 'The protocols from Pubs. **[A]** and **[C]** are...'. At times, the citation itself will be included between brackets, e.g. 'Pub. **[A]** ([2]) contains the first...'. An overview of my publications can be found in a separate list the bibliography.

· Supplementary material to my own publications and other material is presented as a separate list in the bibliography as well . They are referenced with *Sup.* (for *supplementary material*), e.g. '...Python code can be found in Sup. **[sB]**...'.

Notation

Terminology and naming conventions have been adopted from literature as much as possible.

Pauli operators are denoted by X, Y and Z, or P, Q for general (n-qubit) Pauli operators. The notation σ_x etc. is not used in this thesis, except briefly in chapter A.

Classical registers are, whenever possible, denoted with the letters X, Y and Z. At the same time, quantum registers are, whenever possible, denoted with the letters A, B and C.

General quantum states are described by $|\psi\rangle$ and $|\phi\rangle$ if they are pure, or ρ and σ if they are mixed. At times the proper normalization factors will be dropped if it introduces no ambiguity, e.g. in the inline expression $|\mathrm{GHZ}_{\mathbf{P}}\rangle = |0\ldots0\rangle_{\mathbf{P}} + |1\ldots1\rangle_{\mathbf{P}}$, because it reads better without it. Alternatively, the '$\propto$' sign might be used to indicate that a state is merely proportional to another state, e.g. $|B_{00}\rangle \propto |00\rangle + |11\rangle$.

Following [1], with the notation $[n]$ the following set is indicated:

$$[n] = \{1, 2, \ldots, n\}. \tag{1}$$

The notation $\{0,1\}^n$ indicates the set of all possible bit strings of length n.

The operator $\mathbb{I}$ is used to represent the identity operator on any space, whenever context permits. That means that $\mathbb{I}$ can be both the single-qubit operator $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, but also the generalisation to any number of qubits, or other spaces.

For n-qubit operators, the notation from [3] is adopted and adapted. More specifically, superscripts for operators are used to indicate a power, e.g. Z^2 indicates the operator ZZ (which equals $\mathbb{I}$). A superscript of 0 indicates, by convention, the identity operator: $Z^0 = \mathbb{I}$. A superscript including a '$\otimes$' sign is shorthand for an n-fold tensor product: $\mathbb{I}^{\otimes 3} = \mathbb{I} \otimes \mathbb{I} \otimes \mathbb{I}$.

Subscripts for operators are used to indicate (sub)spaces on which the operator acts. For any qubit $a \in [n]$, the notation Z_a denotes the n-qubit operator that acts with the Z operator on qubit a:

$$Z_a = \underbrace{\mathbb{I} \otimes \cdots \otimes \mathbb{I}}_{a-1 \text{ times}} \otimes Z \otimes \underbrace{\mathbb{I} \otimes \cdots \otimes \mathbb{I}}_{n-a \text{ times}} = \mathbb{I}^{\otimes(a-1)} \otimes Z \otimes \mathbb{I}^{\otimes(n-a)}. \tag{2}$$

Note that the total number of qubits n that the operator acts on is only given implicitly. This notation is extended to sets of qubits, and additionally

products are allowed, so that the operator $Z_{\{1,2\}}X_n$ indicates the following n-qubit operator:

$$Z_{\{1,2\}}X_n = Z \otimes Z \otimes \underbrace{\mathbb{I} \otimes \cdots \otimes \mathbb{I}}_{n-3 \text{ times}} \otimes X = Z \otimes Z \otimes \mathbb{I}^{\otimes(n-3)} \otimes X. \tag{3}$$

The orbit of a graph G under local complementations, as introduced in chapter 3, is written $\mathcal{O}(G)$. The local Clifford orbit of a *graph state*, as introduced in chapter 4, is written $\mathcal{O}^{\mathrm{LC}}(|G\rangle)$; to emphasize the difference it is never written $\mathcal{O}^{\mathrm{LC}}(G)$, but only with the full graph state $|G\rangle$ as the parameter. To emphasize the difference between local Clifford and local unitary orbits, an LU-orbit of a graph state $|G\rangle$ is written $\mathcal{O}^{\mathrm{LU}}(|G\rangle)$, again with the *graph state* as the parameter.

The n-element permutation group is denoted $\mathcal{V}_n$. This is not standard notation, which would be S_n, or potentially $\mathcal{S}_n$. However, S_n or $\mathcal{S}_n$, and alternatively P_n or $\mathcal{P}_n$, are all either too close to, or reserved for, other things in this thesis.

List of Figures

3.1 Three graph examples. **39**
3.2 Example of local complementation. **41**
3.3 Example of orbit **42**
3.4 The Bell pair graph state **44**
3.5 The star graph orbit. **46**
3.6 The graph K_3 and three body line graphs **48**
3.7 Example of Z measurement on a graph **50**
3.8 Example of Y measurement on a graph **51**
3.9 Example of X measurement on a graph **54**

4.1 Entanglement class of $|L_4\rangle$ **66**

5.1 Three examples of extraction patterns **78**
5.2 Maximal extraction for $|L_7\rangle$ **80**
5.3 Reduction to the maximal extraction pattern. **84**
5.4 Four- and five-body GHZ state extraction from $|L_7\rangle$. **85**
5.5 Compiled preparation circuit for the linear cluster state **86**
5.6 Lower bound to fidelity of linear cluster and GHZ state **88**
5.7 Alternative lower bound **89**

6.1 Four examples of marginals. **94**
6.2 Three LU-inequivalent graph states **98**
6.3 Two LU-inequivalent graphs with identical l_2 . . . . . **99**
6.4 Two LU-inequivalent graph states with identical T_2^G **101**
6.5 Two graph states from different entanglement classes **106**
6.6 Representatives of three different entanglement classes **108**
6.7 Two LC-inequivalent graph states with identical T_k^G **109**
6.8 The Peterson graph and an LU-inequivalent permutation **110**

8.1 Partitioning of the network **140**
8.2 Visualization of `AME` **141**
8.3 Flowchart of `ACKA` **143**
8.4 Overview of public communication throughout `ACKA` **145**

9.1 Setting of `LinACKA` **159**
9.2 States of the network during `LinACKA` **162**
9.3 Finite key rate r_{net} of `LinACKA` **168**

9.4 Surface plot of finite key rate r_{net} **169**

10.1 State tomography of experimental $|\text{GHZ}_4\rangle$ state **176**
10.2 Success rates for experimental implementation of `ACKA` **178**
10.3 Finite key rate of `LinACKA` as a function of p **179**
10.4 Surface plot of optimal choice of p **180**
10.5 Success rates for the experimental implementation of `LinACKA` . **182**

D.1 Visualization of `NOTIFICATION` **226**
D.2 Visualization of `AME` **228**

F.1 Two exemplary configurations **243**

List of Tables

1.1 Overview of important unitary operations **13**

2.1 The four Bell states **27**
2.2 Measurement of Bell pair **30**
2.3 Bell state measurement for entanglement swapping. **31**

3.1 Generators of the graph state $|K_3\rangle$ **48**

4.1 The number, and average- and maximum sizes of LU-orbits . . . **68**

5.1 Generators of $|L_n\rangle$ state **81**
5.2 Post-measurement state after maximal extraction. **82**

6.1 Performance of T_k **104**
6.2 Performance of l_k **105**
6.3 Performance of t_k **107**

9.1 Selection of steps to perform during `STATE PREPARATION` **161**
9.2 Selection of steps to perform during `GHZ EXTRACTION`. **162**
9.3 Minimum block size to obtain secret key of given length **170**

10.1 Network configurations and experimental results **177**

B.1 Post-measurement state after maximal extraction. **220**

E.1 Different perspective of adversaries in `ACKA` **234**
E.2 Overview of all public communication in `ACKA` **235**

F.1 Local correction to obtain GHZ state. **244**
F.2 Selection of measurement outcomes **244**

Introduction

Physical systems at the atomic scale, like single photons or electrons, are governed by *quantum mechanics* instead of classical physics. The rules that dictate such quantum systems invoke counter-intuitive phenomena, such as the *superposition principle*, which removes the notion that a system is always in one definite configuration, and the *Heisenberg uncertainty principle*, a fundamental limit which states that quantum systems cannot have well-defined values simultaneously for certain pairs of properties, like their position and speed.

Broadly speaking, quantum information science is the field of research that aims to leverage these phenomena by encoding and manipulating *quantum information*: a form of information that can only be represented in quantum mechanics. The idea of using quantum systems as information carriers originated in the 1960s, when Wiesner showed[1] that such quantum information can possess properties that have no counterpart in classical physics [5]. Two decades later, Bennett and Brassard solidified the field when they introduced[2] its first proper application: *quantum key distribution* (QKD) [6] — a concept to be introduced below, that provides strong secure communication in networks.

Quantum information science is a many-faceted field of research, under whose umbrella fall both *quantum computation* [7] and *quantum communication*. Quantum computation leverages quantum information to perform computations in ways that are not possible in classical physics, to open up unmatched possibilities in computation and simulation. Quantum communication considers the distribution of quantum information as signals in a quantum network to realise many realisations of tasks that are either not possible using classical physics, or improve over current methods.

Indeed, such quantum networks are a counterpart or a complementation to classical networks. Their most well-known application is QKD, but there are other applications for quantum networks, including other examples of the subfield of *quantum cryptography* [8, 9] like random number generation [10–12], verified deletion [13, 14], digital signatures [15–17], blind quantum computation [18–21], multiparty computation [22–25] and anonymous communication [26]. Beyond cryptographic tasks, quantum networks can also be used for quantum sensing [27, 28], quantum clock synchronisation [29] and quantum position verification [30, 31].

[1] His research was not accepted for publication until 1983 however, in part because it was initially rejected and Wiesner then did not try any further until much later [4].

[2] Interestingly, again a proper publication didn't follow for another two decades.

Ultimately, an envisioned *quantum internet* [32, 33] would be a world-wide quantum network that connects anyone on earth to perform these applications, creating the opportunity for unprecedented means of communication, security and distributed computation.

The research that I have performed in the field of quantum communication and cryptography has been largely two-fold: multipartite entanglement and anonymous communication. These two topics are introduced separately below, and are each addressed in more detail in their own part of this thesis, specifically parts II and III for multi-partite entanglement and anonymous communication, respectively. A more detailed explanation of the structure of this thesis is presented in the next chapter.

Entanglement

Many of the applications in quantum communication and cryptography utilize one of the quintessential phenomena of quantum mechanics: *entanglement*. Popularly phrased as *spooky action at a distance*[3], entanglement is a direct consequence of the superposition principle and can be understood as a certain quality that the *state* of two or more quantum mechanical systems can have, for which there exists no classical counterpart. The *state* of a (collection of) quantum systems is a description of its relevant information and configuration, and will be addressed in more detail in chapter 1. Entanglement manifests as *correlations* between multiple quantum systems that can not be reproduced by classical physics [34]. These non-classical correlations are leveraged to realise many of the striking results in quantum communication, and they form the basis of the security of most cryptographic applications [1, 17].

Because entanglement is consumed when it is utilized by an application, it can be understood to be a resource in a quantum network. Moreover, it is easily rendered useless by small amounts of noise, and it is difficult to generate and distribute in a network. Thus, good methods of entanglement generation and distribution are paramount to the function of any quantum network; all these aspects of entanglement form an active field of research.

Entanglement was originally conceptualized and studied between two systems, which is generally called *bi-partite* entanglement [35, 36]. The generalization to more than two parties, called *multi-partite* entanglement [37, 38], is a phenomenon that is less well understood. However, it has seen a growing interest in recent years [2, 39–47], because results indicate that it can be used to outperform methods that only rely on bi-partite entanglement in various cryptographic tasks [48–50]. Multi-partite entanglement exists in myriad different forms that can potentially be transformed into each other, and part of

[3] This term was coined by Einstein, although he used it to argue the incompleteness of quantum theory. A long discussion ensued which is beyond the scope of this thesis, but it culminated in the landmark publication of Bell [34] and the associated *Bell tests* that can assert the correctness of the predictions of entanglement.

the theoretical study of multi-partite entanglement is devoted to determine such *equivalence*.

An indispensable tool in the study of multi-partite entanglement is the concept of *graph states*, a special type of quantum state which is multi-partite entangled, and is represented by a mathematical graph [3]. This graph can represent many of the relevant and interesting properties of the associated quantum state, including its form of multi-partite entanglement. Graph states in quantum networks and their entanglement properties are an active field of research [51–55], and my research in multi-partite entanglement has also been governed by graph states. Specifically, it gave the complete characterization of a specific setting were one type of entanglement is to be obtained from another, and also gave methods to compare and categorize all different forms of multi-partite entanglement in graph states. My publications regarding the subject of multi-partite entanglement are Pubs. **[F]** to **[H]**.

Anonymous communication

While entanglement is not a direct application of quantum communication itself, it is an invaluable resource underpinning many quantum communication and cryptography tasks. As mentioned before, the most well-known example of quantum communication and cryptography is QKD, a method to provide — at least in theory — unbreakable public encryption, by using the non-classical properties of quantum information. Modern QKD indeed relies on entanglement for its fundamental security statements.

Encryption allows two parties in a network, colloquially known as *Alice* and *Bob*, to secretly communicate using means of communication that are accessible by anyone else. Alice and Bob wish to uphold this secrecy even in the presence of an *adversary*, usually embodied by the *eavesdropper Eve*, who wishes to *break* the encryption. Unbreakable encryption is possible by means of the one-time-pad (OTP) method [56], but this method relies on a *cryptographic key*: a secret bit string shared between Alice and Bob that no one else has access to. Ultimately, this key needs to be established in such a way that no one else in the network can learn it, because such leakage would make the encryption void. The term *public* in public encryption indicates that Alice and Bob cannot rely on some initial shared secret to establish the key, e.g. by using a shared password.

Strong classical methods to publicly establish a key exist [57, 58] and are widely in use today. However, these methods are only secure assuming certain restrictions — usually phrased as *computational assumptions* — that are put on the adversary. These assumptions have been put under stress by the recent advent of rudimentary functional quantum computers [59, 60]. Indeed, such machines can run Shor's algorithm [61, 62], which is able to break current

classical cryptographic systems[4].

To combat this problem, QKD aims to establish secret keys without relying on any assumptions, thereby providing *unconditional security*[5]. Fundamentally, QKD is a method that allows two parties to establish a key by leveraging the non-classical properties of quantum information; the security of this method is then derived from the physical laws of nature. There is a large body of research that has been exploring the idea since its conception, and recently it has been argued that indeed a practical advantage over classical methods is within reach [63]. A generalization of QKD is *conference key agreement* (CKA), that allows more than two parties in a network to establish secret keys.

Beyond providing security in communication, quantum communication can hide the *identity* of the involved parties from the rest of the network, thereby providing *anonymity*. My research has explored *anonymous conference key agreement* (ACKA), the combination of anonymity and conference key agreement. The research provided the first protocol that performs ACKA, and various additional protocols that improved upon the first in different settings. Beyond theoretical analyses, experimental implementations were realised. These realisations were not performed by me, but I did perform or aided in the theoretical analysis and post-processing. My publications regarding the subject of anonymity are Pubs. **[A]** to **[E]**.

Topics not discussed in this thesis

Although most of the work that I have conducted in the scope of my PhD is presented in this thesis, there are a few other topics I have worked on that are worth mentioning here. They are either very recently finished or ongoing projects, unsuccessful projects, or topics that have no direct merit for scientific publication. In no particular order, they are:

- The concept of *public quantum cryptography*, which I explored together with Alex Grilo at Sorbonne Université in Paris, France. The idea that we worked on was promising but in an early stage, and when two very similar ideas where published [64, 65], we did not pursue it any further.
- My work for the Quantum Internet Alliance [33], a European-wide project to realize the worlds first quantum internet, built in Europe.
- Pub. **[H]**, which is only very recently available for preprint. It is closely related to the contents of Pub. **[G]**, to be discussed in chapter 6, and will be briefly mentioned there.

[4] In fact, the boom in popularity of quantum computation that was provoked by the publication of Shor can be seen as the *first* quantum revolution (see preface).

[5] Whether QKD truly provides unconditional security is a hotly debated topic, which will be addressed in more detail in chapter 7.

- The `graphstabilizer` Python package (Sup. **[sC]**), a set of tools to work with graph states, the topic of chapter 3, that I developed while working on these states. It can also be used to plot graphs; all figures of graphs in this thesis were made with it.
- An ongoing project with colleagues, and researchers from Sorbonne Université in Paris, France to combine the concept of *anonymity* in quantum networks (to be discussed in part III), with the concept of *privacy* in networked quantum sensing [66, 67].

Structure of this thesis

The main body of this thesis consists of three parts, each with multiple chapters. Furthermore, these chapters each have their own introduction, and their own conclusion, unless specifically stated otherwise.

My own research is presented in the second and third part, each on a separate field of study. They are similarly structured: both start with a chapter that introduces and explains the relevant literature, followed by multiple chapters that each present one or more of my publications.

Each of these chapters on my own publications contain a brief discussion of the associated publication in their introduction. Moreover, their conclusions contain some of the discussions that were presented in the original publications, including ideas for further research.

A brief description of the remainder of this thesis is as follows:

Part I - Mathematical Properties of Quantum Networks

Part I introduces the relevant background information that is needed to present and discuss the results from the rest of the thesis. Specifically, chapter 1 introduces the basic concepts of classical- and quantum information science that are relevant to my publications. This includes the basic notions of quantum states, operations and measurements, quantum entanglement, and (both classical and quantum) entropies.

Two special topics require their own chapter. Chapter 2 presents the so-called *stabilizer formalism*, a mathematical framework and theory that efficiently describes a versatile set of quantum states. Afterwards, chapter 3 introduces the concept of *graph states*, a strong graphical tool to represent and study many interesting aspects of the stabilizer formalism, including its entanglement properties. They are foundational to the study of quantum networks, and parts II and III heavily rely on them.

The reader that is familiar with these concepts is likely safe to skip their respective chapters. However, chapters 2 and 3 contain some topics and details that might still prove unfamiliar; for easy reference these topics are explicitly stated in the introductions of these chapters.

Part II - Multi-partite Entanglement in Quantum Networks

Part II concerns the *distribution* and *characterisation* of multi-partite entanglement in quantum networks. There are myriad forms of entanglement, but two quantum states may be said to have the equivalent form of entanglement, even though they are distinct quantum states. Chapter 4 makes this notion

of *equivalence* more precise, and introduces the relevant concepts and results from literature.

Chapter 5 presents the contents of Pub. **[F]** ([68]). It is studied if, in a networked scenario, a specific type of quantum state can be obtained from another quantum state. It gives a complete characterization of when this is, and is not possible.

The last chapter of the part, chapter 6, is associated with Pub. **[G]** ([55]). It takes a more abstract approach than the previous chapter, and provides methods to characterize the form of entanglement for a given quantum state, and additionally provides methods to compare two or more quantum states regarding their equivalence.

Part III - Anonymous Conference Key Agreement

Part III is on a more operational aspect of quantum communication. Specifically, it regards the topic of *anonymous conference key agreement*, a specific quantum cryptographic task. *Conference key agreement* (*CKA*) is a generalization of QKD to more than two parties. Chapter 7 concerns both QKD and CKA, and explains the relevant literature to obtain modern, strong encryption through QKD. As its title suggests, this part is on *anonymous* conference key agreement (ACKA), in which the parties not only communicate privately, but also remain anonymous, in the sense that no one else in the network knows their identities. The concept of anonymity is also presented in chapter 7. Note that facets of anonymity were originally presented as new research in Pubs. **[A]** and **[C]**, even though they are included in the introductory chapter of this part.

Chapter 8 presents the contents of Pubs. **[A]** and **[C]** ([2, 48]), that presented protocols to perform ACKA in a *star network*. These publications were the first to introduce such protocols, but make use of a somewhat stringent network topology.

Chapter 9 presents the contents of Pub. **[D]** ([46]), that presented an ACKA protocol in a *linear network*, which is a less stringent network topology.

To complement the theoretical presentations of chapters 8 and 9, chapter 10 presents the contents of Pubs. **[B]** and **[E]** ([45, 47]). These two publications presented an experimental proof-of-concept realisation of the protocols of Pubs. **[A]** and **[D]**. The actual experiments were not performed by me, but the analysis and post-processing I did perform; these are presented in the chapter.

Conclusion, Bibliography and Appendices

This thesis is concluded in chapter 11. Various ideas for further and future research that are not restricted to any single of my publications are presented in the chapter thereafter.

The bibliography is included after the conclusion, and my publications are listed separately.

Various prolonged discussions, proofs and other sections have been deferred to the appendices.

PART I

Mathematical Properties of Quantum Networks

1

Mathematical preliminaries

This chapter introduces and defines some basic concepts in quantum information theory. It explains only those concepts and properties that are directly of use in the rest of this thesis, and by no means aims to provide a comprehensive introduction; the quintessential introduction to quantum- computation and information is the seminal book by Nielsen & Chuang [7]. Books more focussed at quantum information science are the one by Watrous [36] and by Wilde [69]. For quantum communication specifically, a good introduction can be found in the book by Khatri and Wilde, available for preprint [70]. Both quantum computation and communication thrive from the concept known as *entanglement*, which is comprehensively studied in [35, 71]. Finally, the book by Vidick and Wehner [72] provides an introduction specifically to quantum cryptography.

A central cornerstone of quantum communication is formed by the *Pauli matrices* and *Pauli group*. They are introduced in sec. 1.1, where the relevant of their many useful properties are explained as well.

The *quantum state*, the mathematical description of the relevant configuration of a quantum mechanical system, is the main ingredient to any quantum computation, or quantum communication protocol. Unless explicitly stated otherwise, in this thesis only *qubits* are considered: the most basic form of a quantum state with only two levels of freedom, so that they are the direct quantum mechanical counterpart to the classical bit. Section 1.2 introduces

them, where additionally various concepts and conventions are defined.

To utilize quantum states one must perform *operations* on them. More specifically, *unitary* operations can change quantum states to other quantum states, and *measurements* are operations that extract classical data from quantum states in the form of measurement outcomes. Both these types of operations are discussed in sec. 1.3.

In both classical and quantum information science, the concept of *entropy* is an important tool to determine many qualities of random processes. They come in different forms and are indispensable in, among other applications, the study of quantum cryptography. Those entropies that are used in the rest of this thesis are introduced in sec. 1.4.

One of the defining qualities of quantum states is that they can be *entangled*. Such *entangled* states show behaviour that can not be mimicked by classical systems, and this behaviour is leveraged by many of the applications in quantum communication and computation. Entanglement is introduced in sec. 1.5, where additionally the (arguably) most important and fundamental entangled state is defined: the *Bell* or *EPR* pair.

An important subset of all possible quantum operations are the *Clifford operations*. These, and the associated *Clifford group*, are introduced in sec. 1.6. The section additionally introduces the notion of *local unitary operations*, and the interplay between the two: the *local Clifford operations*. All these concepts play an important role in the study of entanglement, which will be discussed in part II. As this chapter presents the basics of quantum information science, the familiar reader may feel free to skip this chapter.

1.1 The Pauli group

The three *Pauli* matrices X, Y and Z, named after the famous physicist, are operators that play an integral role in quantum computation and communication. Together with the identity operator $\mathbb{I}$, they form a basis of the space of 2×2 matrices, and are defined as:

$$\mathbb{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \qquad X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix},$$
$$Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \qquad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \tag{1.1}$$

Up to a phase, the Pauli operators are related to each other by multiplication:

$$\begin{aligned} YZ &= iX, \\ ZX &= iY, \\ XY &= iZ. \end{aligned} \tag{1.2}$$

Through the tensor product, they can be extended to form the *Pauli group* $\mathcal{P}_n$.

Definition 1. *The n-qubit* Pauli group $\mathcal{P}_n$ *is the set of all n-fold tensor products of the single-qubit Pauli operators:*

$$\mathcal{P}_n = \{1, -1, i, -i\} \cdot \langle \mathbb{I}, X, Y, Z \rangle^{\otimes n}, \tag{1.3}$$

where the phases $\{1, -1, i, -i\}$ *have been introduced so that the set is closed. It is straightforward to verify that the Pauli group* $\mathcal{P}_n$ *indeed forms a group.*

In the remainder of this thesis, the term *Pauli operator* will be reserved for the extensions, i.e. the elements of the Pauli group $\mathcal{P}_n$, and not just the operators from (1.1) (unless explicitly stated otherwise). Rather, the operators from (1.1) will be explicitly referred to with X, Y and Z whenever possible.

Any element $P \in \mathcal{P}_n$ can be written as a tensor product of single-qubit Pauli operators $P_j \in \mathcal{P}_1$:

$$P = \{\pm 1, \pm i\} \cdot \bigotimes_{j \in [n]} P_j. \tag{1.4}$$

In this thesis, it can usually be assumed that the phase of a Pauli operator is either $+1$ or -1, unless explicitly stated otherwise. Moreover, when it is not important, the phase will be dropped.

The Pauli group $\mathcal{P}_n$ has many interesting and useful properties; the remainder of this section will discuss those properties that are applicable or relevant to the rest of this thesis. Additionally, some other associated concepts that are useful in later chapters are defined.

The Pauli operators are both unitary and Hermitian, which implies that they are their own inverse. Moreover, the Pauli operators either *commute* or *anti-commute*. More specifically, for two Pauli operators $P, Q \in \mathcal{P}_n$, either of the following two equations hold:

$$\begin{aligned} [P,Q] &= PQ - QP = 0, \\ \{P,Q\} &= PQ + QP = 0. \end{aligned} \tag{1.5}$$

For any Pauli operator $P = \{\pm 1, \pm i\} \cdot \bigotimes_{j=1}^{n} P_j$, the set of tensor factors on which it acts non-trivially (i.e. those j for which $P_j \in \{X, Y, Z\}$, and not $\mathbb{I}$) is called its *support*:

$$\mathrm{supp}(P) = \{j \in [n] = \{1 \dots n\} | P_j \neq \mathbb{I}\}. \tag{1.6}$$

The *weight* $\mathrm{w}(P)$ of a Pauli operator $P \in \mathcal{P}_n$ is the number of elements in its support: $\mathrm{w}(P) = |\mathrm{supp}(P)|$. Any n-qubit Pauli operator that has $\mathrm{w}(P) = n$ is said to have *full weight*.

The trace of the 2×2 identity operator $\mathbb{I}$ equals two, and the single-factor Pauli operators X, Y and Z have trace zero. Using the identity $\operatorname{tr}\left[A \otimes B\right] = \operatorname{tr}\left[A\right] \operatorname{tr}\left[B\right]$, it follows that, up to a phase, for any $P \in \mathcal{P}_n$ it holds that:

$$\operatorname{tr}\left[P\right] = \begin{cases} 2^n & \text{when } P = \mathbb{I}, \\ 0 & \text{otherwise.} \end{cases} \tag{1.7}$$

This can be generalized to *partial* traces. The partial trace over the last $n-k$ tensor factors of the Pauli operator $P = \bigotimes_{j=1}^{n} P_j$ is only non-zero when its support $\operatorname{supp}(P)$ is contained in all factors that are not traced out:

$$\operatorname{tr}_{k+1,\ldots,n}\left[P\right] = \begin{cases} 2^{n-k} \bigotimes_{j=1}^{k} P_j & \text{when } \operatorname{supp}(P) \subseteq \{1 \ldots k\}, \\ 0 & \text{otherwise.} \end{cases} \tag{1.8}$$

Partial traces over any other subset follow similarly. Note that any phase $\{\pm 1, \pm i\}$ has been omitted from (1.7) and (1.8).

Pauli eigenspaces

The eigenspaces of Pauli operators will play a central role in chapter 2, and therefore some useful properties of the associated projectors are introduced. Because the Pauli operators are both Hermitian and unitary, they have only a $+1$ and a -1 eigenspace. The spectral theorem implies that any Pauli operator $P \in \mathcal{P}_n$ can thus be written as:

$$P = \Pi_{+1}^{P} - \Pi_{-1}^{P}, \tag{1.9}$$

where Π_{+1}^{P} and Π_{-1}^{P} are the projectors upon the $+1$ and -1 eigenspaces of P:

$$\Pi_{+1}^{P} = \frac{\mathbb{I} + P}{2}, \quad \Pi_{-1}^{P} = \frac{\mathbb{I} - P}{2}. \tag{1.10}$$

Because the dimension of an eigenspace is equal to the trace of its projector, it follows that both eigenspaces have equal dimensions, namely 2^{n-1}.

If a Pauli operator $P \in \mathcal{P}_n$ commutes with another Pauli operator $Q \in \mathcal{P}_n$, then P commutes with the $+1$ and -1 eigenspace projectors of Q as well. This is shown by the following equation for the $+1$ eigenspace projector:

$$P\Pi_{+1}^{Q} = P\left(\frac{\mathbb{I} + Q}{2}\right) = \frac{P + PQ}{2} = \frac{P + QP}{2} = \left(\frac{\mathbb{I} + Q}{2}\right) P = \Pi_{+1}^{Q} P. \tag{1.11}$$

The case for the -1 eigenspace projector follows similarly.

A separate, useful result regarding the $+1$ eigenspaces of two commuting Pauli operators P and Q, is that their overlap is contained in the $+1$ eigenspace of the Pauli operator PQ. More specifically, let Π_{+1}^{P} and Π_{+1}^{Q} be the

+1 eigenspace projectors for two commuting Pauli operators P and Q. Additionally, let $\Pi_{+1}^{(PQ)}$ be the projector for the +1 eigenspace of the product PQ. The overlap of the subspaces associated with Π_{+1}^{P} and Π_{+1}^{Q} is then always contained in the +1 eigenspace of PQ, which the following calculation shows:

$$\begin{aligned}\Pi_{+1}^{(PQ)}\Pi_{+1}^{P}\Pi_{+1}^{Q} &= \frac{1}{8}(\mathbb{I}+PQ)(\mathbb{I}+P)(\mathbb{I}+Q)\\ &= \frac{\mathbb{I}}{8}(\mathbb{I}+P+Q+PQ)+\frac{PQ}{8}(\mathbb{I}+P+Q+PQ)\\ &= \frac{2}{8}(\mathbb{I}+P+Q+PQ) = \Pi_{+1}^{P}\Pi_{+1}^{Q}.\end{aligned} \tag{1.12}$$

This result shows that the shared +1 eigenspace of the operators P, Q and PQ is determined by P and Q alone, a fact that will be important in chapter 2.

1.2 | Quantum states

A quantum state, denoted $|\psi\rangle$, is the mathematical description of the configuration of a quantum mechanical system. In its most basic form, such a quantum mechanical system is a two-level system, with the two levels usually labelled as $|0\rangle$ and $|1\rangle$. Such a system is called a *qubit*.

One defining feature of quantum states is that they can be in a *superposition* of the two basis states: a linear combination with coefficients $\alpha, \beta \in \mathbb{C}$. In its most general form, the state $|\psi\rangle$ of a qubit can thus be written as:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,. \tag{1.13}$$

The coefficients are subject to the *normalization condition*:

$$|\alpha|^2 + |\beta|^2 = 1. \tag{1.14}$$

Mathematically, a quantum state is an element[1] of a Hilbert space $\mathcal{H}_2$; the basis $\{|0\rangle, |1\rangle\}$ which is used in (1.13) is called the *computational basis*, and the two states are the +1 and −1 eigenstates of the Pauli Z operator, respectively.

Because a state is a vector in a Hilbert space $\mathcal{H}_2$, it can be expressed in any other basis of the space. For instance, in the basis $\{|+\rangle, |-\rangle\}$, where the states $|+\rangle$ and $|-\rangle$ are the +1 and −1 eigenstates of the (single-qubit) Pauli X

[1] Two quantum states that differ only by a *global* phase, e.g. $|0\rangle$ and $-|0\rangle$, are physically identical. Therefore, such a global phase is physically irrelevant, and one can even define a quantum state to be a *ray* in a Hilbert space. Alternatively, a quantum state can be defined as an element of a *complex projective Hilbert space*. Another approach, adopted by e.g. [36], is to define quantum states purely in terms of density matrices, introduced below. However, for this thesis no such extra specification is necessary.

operator (therefore known as the X-basis, and additionally as the *Hadamard* basis):

$$|+\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right), \qquad |-\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right). \tag{1.15}$$

The state $|\psi\rangle$ expressed in this basis then becomes:

$$\begin{aligned} |\psi\rangle =& \alpha\,|0\rangle + \beta\,|1\rangle \\ =& (\alpha+\beta)\,|+\rangle + (\alpha-\beta)\,|-\rangle\,. \end{aligned} \tag{1.16}$$

Like the case for the X-basis, the $+1$ and -1 eigenstates of the Pauli Y operator, labelled $|+i\rangle$ and $|-i\rangle$, form the Y-basis:

$$|+i\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + i\,|1\rangle\right), \qquad |-i\rangle \;\; = \frac{1}{\sqrt{2}}\left(|0\rangle - i\,|1\rangle\right). \tag{1.17}$$

For any state $|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$, its *dual* $\langle\psi|$ is a map $\mathcal{H} \to \mathbb{C}$. For the contents of this thesis, the state $|\psi\rangle$ can be viewed as a column vector $|\psi\rangle = \begin{bmatrix}\alpha\\ \beta\end{bmatrix}$, and its dual can be viewed as a row vector $\langle\psi| = \begin{bmatrix}\alpha^* & \beta^*\end{bmatrix}$, where α^* denotes the complex conjugate of α, and likewise for β^*.

For states $|\psi_1\rangle = \alpha_1\,|0\rangle + \beta_1\,|1\rangle$ and $|\psi_2\rangle = \alpha_2\,|0\rangle + \beta_2\,|1\rangle$, the expression $\langle\psi_1|\psi_2\rangle$ then means:

$$\langle\psi_1|\psi_2\rangle = \begin{bmatrix}\alpha_1^* & \beta_1^*\end{bmatrix}\begin{bmatrix}\alpha_2\\ \beta_2\end{bmatrix} = \alpha_1^*\alpha_2 + \beta_1^*\beta_2, \tag{1.18}$$

while the expression $|\psi_1\rangle\langle\psi_2|$ means:

$$|\psi_2\rangle\langle\psi_1| = \begin{bmatrix}\alpha_2\\ \beta_2\end{bmatrix}\begin{bmatrix}\alpha_1^* & \beta_1^*\end{bmatrix} = \begin{bmatrix}\alpha_1^*\alpha_2 & \beta_1^*\alpha_2\\ \alpha_1^*\beta_2 & \beta_1^*\beta_2\end{bmatrix}. \tag{1.19}$$

Note that, for two states $|\psi\rangle$ and $|\phi\rangle$ that only differ in a *global phase* (i.e. $|\phi\rangle = e^{i\phi}\,|\psi\rangle$), the expressions $|\psi\rangle\langle\psi|$ and $|\phi\rangle\langle\phi|$ coincide. These global phases are physically irrelevant.

Eq. (1.19) allows the eigenspace projectors of the X, Y and Z operators to be written in terms of their eigenstates. It follows that the Pauli operators X, Y and Z can be written as:

$$\begin{aligned} X &= |+\rangle\langle+| - |-\rangle\langle-|\,, \\ Y &= |+i\rangle\langle+i| - |-i\rangle\langle-i|\,, \\ Z &= |0\rangle\langle0| - |1\rangle\langle1|\,. \end{aligned} \tag{1.20}$$

Multiple qubits states

The states of multiple quantum systems can be combined using the *tensor product*. More specifically, suppose that two qubit systems A and B are in the state $|\psi_A\rangle = a_0 |0\rangle_A + a_1 |1\rangle_A$ and $|\psi_B\rangle = b_0 |0\rangle_B + b_1 |1\rangle_B$, where the underscores A and B indicate the systems. The state $|\psi\rangle_{AB}$ of the combined system AB is then an element of the compound Hilbert space $\mathcal{H}_{AB} = \mathcal{H}_A \otimes \mathcal{H}_B$:

$$\begin{aligned} |\psi\rangle_{AB} &= |\psi_A\rangle \otimes |\psi_B\rangle \\ &= (a_0 |0\rangle_A + a_1 |1\rangle_A) \otimes (b_0 |0\rangle_B + b_1 |1\rangle_B) \\ &= \sum_{i,j\in\{0,1\}} a_i b_j |i\rangle_A \otimes |j\rangle_B \end{aligned} \tag{1.21}$$

This is usually simplified by dropping the explicit '$\otimes$' signs. Moreover, when context permits, the description of the systems is dropped as well:

$$|\psi\rangle_{AB} = a_0 b_0 |00\rangle + a_0 b_1 |01\rangle + a_1 b_0 |10\rangle + a_1 b_1 |11\rangle . \tag{1.22}$$

This procedure can be generalized to describe the state of any number of qubits. Note that there are generally 2^n coefficients necessary to describe the state of an n-qubit system, because the qubits can be in a superposition.

Mixed states

If there exists statistical uncertainty regarding the state of a quantum system, it is described by a statistical mixture of quantum states. The mathematical method to do so is the *density matrix*, a $2^n \times 2^n$ matrix. A density matrix, usually written ρ or σ, must be positive semidefinite, and is subject to the *normalization condition* $\operatorname{tr}[\rho] = 1$. As a direct consequence of the spectral theorem, ρ can be written in its *spectral decomposition*:

$$\rho = \sum_{i=1}^{2^n} \lambda_i |\psi_i\rangle\langle\psi_i| , \tag{1.23}$$

where the λ_i's are the eigenvalues of ρ, and the normalization condition implies $\sum_i \lambda_i = 1$. Note that one or more λ_i's may be zero, so that the rank $\operatorname{rnk}(\rho)$ of ρ may not be 2^n.

In the special case that ρ has rank $\operatorname{rnk}(\rho) = 1$, it can be written as $\rho = |\psi\rangle\langle\psi|$, for some quantum state $|\psi\rangle$. It is then called *pure*, and usually it is described as $|\psi\rangle$ (instead of $|\psi\rangle\langle\psi|$ or ρ). Any quantum state ρ that is not pure is called *mixed*.

To emphasize the difference with (statistical) mixtures, a pure quantum state that is in a superposition is often said to be *coherent*, or be in a *coherent superposition*.

Classical states

In the context of (quantum) information science, a classical system is often called a *classical register* or just *register*. In this thesis such a classical register is an n-bit system, whose state can be any of the 2^n bit-strings, unless explicitly stated otherwise. Because it is classical, it can not be in a coherent superposition. Nevertheless, the state of the register can be a statistical mixture described by a probability distribution p over all 2^n possible states of the register. Sometimes it is useful to represent this (classical) state as a diagonal density matrix $\rho_{\text{classical}}$:

$$\rho_{\text{classical}} = \sum_{i \in \{0,1\}^n} p(i) \, |i\rangle\langle i| \, . \tag{1.24}$$

Each time it is used it should be made clear, either explicitly or through context, that a classical system is indeed a classical system, and thus cannot have off-diagonal elements, or be in a coherent superposition.

Reduced states

For a system of n qubits in the state ρ, one may be interested in the state of only a subset $M \subset [n]$ of the qubits. The density matrix ρ_M that describes this state is called the *marginal-* or *reduced* state of ρ on M, or just the *marginal*[2]. It can be computed from ρ by tracing over all other qubits:

$$\rho_M = \text{tr}_{M^\perp} [\rho] \, , \tag{1.25}$$

where $M^\perp = [n] \setminus M$ is the complement of M. Sometimes, when context permits, the M is dropped, and the reduced state is just called the *marginal*.

For any state ρ defined on some system A, a *purification* of ρ is a pure state $|\psi\rangle_{AB} \in \mathcal{H}_A \otimes \mathcal{H}_B$, for some extra system B, such that the reduced state on A equals ρ:

$$\rho = \text{tr}_B \left[|\psi\rangle\langle\psi|_{AB} \right] . \tag{1.26}$$

A purification is not unique, but many of its properties are the same for every choice of purification [7].

Distinguishing quantum states

Although, for instance, the basis states of the Pauli Z operator are orthogonal, two arbitrary quantum states ρ and σ will generally have some

[2] Because it is easy to do so, whenever context permits I will use the word 'marginal' for both the reduced state ρ_M and for the selection of qubits M. Moreover, when it makes sense, the term *k-body marginal* refers to a marginal with k elements.

non-zero overlap. In this sense, quantum states can be understood to have a *distance* between each other. There are two standard notions of distance between quantum states: the *fidelity* and the *trace distance* [7, 36]. Both represent a notion of how *close* two states can be, so that close states are less distinguishable than states that are further apart.

The fidelity $F(\rho, \sigma)$ is defined as [36]:

$$F(\rho, \sigma) = \left\| \sqrt{\rho}\sqrt{\sigma} \right\|_1 = \operatorname{tr}\left[\sqrt{\sigma}\rho\sqrt{\sigma} \right] . \tag{1.27}$$

If at least one of the states is pure, e.g. $\rho = |\psi\rangle\langle\psi|$, the fidelity simplifies to:

$$F(|\psi\rangle , \sigma) = \langle\psi|\sigma|\psi\rangle . \tag{1.28}$$

In general $0 \leqslant F(\rho, \sigma) \leqslant 1$, where $F(\rho, \sigma) = 0$ indicates that the states are completely orthogonal, and $F(\rho, \sigma) = 1$ indicates that the states are identical (up to a global phase).

Like the fidelity, the trace distance $D_{\mathrm{tr}}(\rho, \sigma)$ is defined on two states ρ and σ:

$$D_{\mathrm{tr}}(\rho, \sigma) = \frac{1}{2} \left\| \rho - \sigma \right\|_1 . \tag{1.29}$$

Again, in general $0 \leqslant D_{\mathrm{tr}}(\rho, \sigma) \leqslant 1$, but now $D_{\mathrm{tr}}(\rho, \sigma) = 0$ indicates that the states are identical, and $D_{\mathrm{tr}}(\rho, \sigma) = 1$ indicates that the states are completely orthogonal.

The trace distance has an important operational interpretation in terms of the power to distinguish two states. If any generalized measurement, to be defined in sec. 1.3, is performed on a mixture of the two states, the measurement outcomes' ability to distinguish the two states is bounded by the trace distance. This statement will be made more precise by (1.38) in sec. 1.3.

1.3 | Operations on qubits

A quantum system can be acted upon, so that its state ρ is transformed into some other state σ. There exists a rich theory of the types of transformations that are possible, known as *completely positive and trace preserving* or CPTP maps [7, 36, 69]. Such a CPTP map Λ, also referred to as a *quantum channel*, is a linear map:

$$\sigma = \Lambda(\rho). \tag{1.30}$$

The condition that the map is *completely positive* and *trace preserving* guarantees that the output σ is a correctly defined quantum state, even if the state ρ was part of a larger state. Quantum information theory is largely concerned with the study of quantum channels and how they can affect quantum states [36, 69].

An important subclass of all CPTP maps are the *unitary evolutions* or *unitary rotations*. These are described by *unitary operators*, i.e. elements of

the *unitary group* $\mathcal{U}_n$. Here, the n indicates that the unitary applies to n qubits, so that it is a $2^n \times 2^n$ matrix. More specifically, a *unitary operation* maps a quantum state ρ to a state σ:

$$\rho \to \sigma = U\rho U^\dagger. \tag{1.31}$$

In the case that $\rho = |\psi\rangle\langle\psi|$ is pure, the output state $\sigma = |\phi\rangle\langle\phi|$ is pure as well, and the transformation is written as:

$$|\psi\rangle \to |\phi\rangle = U|\psi\rangle\,. \tag{1.32}$$

In the context of quantum computation, unitary operators acting on qubits are also called *gates*, a convention which is adopted in quantum communication.

As a general rule of thumb, in quantum computation the goal is to implement unitary operations, while more general CPTP maps and mixed states are usually unwarranted. When a state is mixed in a quantum computation, it usually means that it is *noisy*, which has to be addressed by *quantum error correction* [73, 74] and *fault tolerance* [75].

On the other hand, statistical mixtures play an integral role in quantum communication (consider e.g. cryptography, where an encryption key must be completely unknown and random to an adversary). Thus, CPTP maps and mixed states are prevalent in this field, although CPTP maps are only used implicitly in this thesis.

As noted before, the Pauli operators are unitary. The X and Z are known as the *bit-flip* and *phase-flip* operators, respectively, because of their action on computational basis states. Some other important unitary operators are presented in **Tab.** 1.1, and introduced in further detail below.

The operator H is called the *Hadamard* operator, and swaps between the computational and Hadamard basis. As their notation suggests, the operators $\sqrt{X}$, $\sqrt{Y}$ and $\sqrt{Z}$ are operators that square to X, Y and Z, respectively. Using (1.20), $\sqrt{X}$ can be computed as:

$$\sqrt{X} = \sqrt{1}\,|+\rangle\langle+| + \sqrt{-1}\,|-\rangle\langle-| = |+\rangle\langle+| + i\,|-\rangle\langle-|\,, \tag{1.33}$$

and $\sqrt{Y}$ and $\sqrt{Z}$ follow similarly. The $\sqrt{Z}$ operator is sometimes referred to as the S gate. The T gate is the (positive) square root of the $\sqrt{Z}$ gate, and plays an important role in quantum computation [7] because of its ties to fault-tolerance [76] (see also sec. 2.5). The $P(\phi)$ gate, known as the *phase gate*, is a generalization of the Z, S and T gate to arbitrary phases.

The C_X and C_Z gates act on two qubits and are therefore *two-qubit gates*. Moreover, they are *conditional* gates, where its action on one of the qubits (the *target*) depends on the state of the other (the *control*). They can be written as a sum of tensor factors:

$$\begin{aligned} C_X^{1\to 2} &= |0\rangle\langle 0| \otimes \mathbb{I} + |1\rangle\langle 1| \otimes X, \\ C_Z^{1\to 2} &= |0\rangle\langle 0| \otimes \mathbb{I} + |1\rangle\langle 1| \otimes Z, \end{aligned} \tag{1.34}$$

$$H = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad \sqrt{X} = \frac{1}{2}\begin{bmatrix} 1+i & 1-i \\ 1-i & 1+i \end{bmatrix}$$

$$\sqrt{Y} = \frac{1}{2}\begin{bmatrix} 1+i & -1-i \\ 1+i & 1+i \end{bmatrix} \qquad \sqrt{Z} = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$$

$$T = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{bmatrix} \qquad P(\phi) = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{bmatrix}$$

$$C_X^{1\to 2} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad C_Z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

Table 1.1: An overview of important unitary operations. The first three rows consists of single-qubit operators, that change the state of a single qubit. The last row contains two-qubit gates, that act on the composition of two qubits.

which shows why they are referred to as the *controlled-X* and *controlled-Z* gates, respectively. The superscript indicates the control *towards* the target qubit; because the C_Z gate is symmetric (i.e. $C_Z^{1\to 2} = C_Z^{2\to 1}$), its superscript will be dropped or written as e.g. $C_Z^{(1,2)}$ in the remainder of this thesis.

Measurements

Quantum measurements forms a rich topic with many different formulations [7, 36]. For most purposes of this thesis, a measurement of a quantum system in the state ρ can be understood as a *PVM* or *projector-valued measurement*. Such a measurement results in a *measurement outcome* m, randomly drawn from the set of possible outcomes $\mathcal{M}$; w.l.o.g. the set $\mathcal{M}$ can be understood to consist of only real-valued numbers. With every possible outcome $x \in \mathcal{M}$, a projection operator Π_x is associated. These projection operators are also known as the *measurement operators*, and they must obey the *completeness relation* $\sum_{x\in\mathcal{M}} \Pi_x = \mathbb{I}$.

The probability $\Pr(m = x)$ that the measurement results in the outcome $m = x$ can be calculated by the *Born rule* [7]:

$$\Pr(m = x) = \operatorname{tr}\left[\Pi_x \rho\right]. \tag{1.35}$$

In general, the state of the quantum system is non-trivially affected by a measurement. When the outcome $m = x$ is obtained, the state *collapses* to

the *post-measurement state*:

$$\frac{\Pi_x \rho \Pi_x}{\operatorname{tr}\left[\Pi_x \rho\right]^2}, \tag{1.36}$$

where the denominator is there to ensure that the post-measurement state is properly normalized.

The *observable* $O = \sum_{x \in \mathcal{M}} x \Pi_x$ can be used to calculate the *expectation value* $\mathbb{E}(O)$ of the measurement:

$$\mathbb{E}(O) = \operatorname{tr}\left[O\rho\right]. \tag{1.37}$$

In the special case that the measurement operators are the eigenspace projectors of a Pauli operator, e.g. Π^Z_{+1} and Π^Z_{-1} for the operator Z, the possible measurement outcomes are taken to be the eigenvalues $+1$ and -1, respectively. Such a *Pauli-basis* measurement results in the outcome[3] $m = +1$ or $m = -1$, with the probabilities still dictated by (1.35).

Especially common are measurements in the X-, Y- and Z-basis, where the measurement operators are their eigenspace projectors (see (1.10)) and the outcomes are their eigenvalues $+1$ or -1. Note that a measurement of a general n-qubit Pauli operator P is possible as well, where the outcome is still $+1$ or -1, i.e. either of its eigenvalues. However, the post-measurement state is then not collapsed onto a basis state, but onto the eigenspace of P associated with the measurement outcome (see (1.10)).

PVMs are not the most general description of quantum measurements. A more complete description is given by a *positive operator valued measurement* or *POVM*, where the measurement operators are not just projectors but replaced by positive semidefinite operators $\{E_x\}$ s.t. $\sum_x E_x = \mathbb{I}$. A POVM on a quantum system A can be understood as a PVM on a compound quantum system AB, i.e. where the system A has been *extended* by an extra system B that is usually referred to as the *environment*. These *generalized measurements* [7, 36] will show up in certain definitions in this thesis, but calculations with them are not necessary.

Generalized measurements can be used to make the operational meaning of the trace distance more precise, as e.g. by theorem 9.1 from [7]. More specifically, let $\{E_m\}$ be a POVM, and let $p_m = \operatorname{tr}\left[\rho E_m\right]$ and $q_m = \operatorname{tr}\left[\sigma E_m\right]$ create the probability distributions p and q of the measurement outcomes on

[3] Instead of labelling the outcomes of Pauli measurements with their eigenvalues $+1$ and -1, they are often labelled with 0 and 1, especially in quantum networking protocols. I like to use both in different settings: $+1$ and -1 are slightly more intuitive because they are the actual eigenvalues of the measurement observable, but 0 and 1 better reflect that it's a binary outcome encoded by a single bit — this is a very useful representation in e.g. networking protocols. For this reason, physicist generally tend to use the first representation, whereas in computer science the second representation is more prevalent. In later chapters I will generally use $\{0, 1\}$, and to (hopefully) limit confusion and ambiguity I will always explicitly write the '+' in '+1' for the outcome in the $\{+1, -1\}$ representation, so that '1' is reserved for the outcome in the $\{0, 1\}$ representation.

the two states resulting from this POVM. Then the total variational distance of these measurement outcomes is bounded by the trace distance:

$$D(p_m, q_m) \leqslant D_{\mathrm{tr}}(\rho, \sigma). \tag{1.38}$$

This bound is a special case of a more general theorem, called the *Holevo-Helstrom theorem* [36].

1.4 | Entropies

Entropies are useful quantities that are used throughout this thesis. Many different interpretations of what exactly an entropy is exist, but in general they are a measure of *randomness* that probability distributions can have. There does not exist one unique entropy, but there are multiple related concepts. They are all sometimes referred to as *entropic measures*.

The most foundational entropy is the *Shannon entropy*, named after Shannon who introduced it in his seminal paper [77]. It is either defined on a probability distribution, or on a classical register X with a state described by such a probability distribution.

Definition 2. *Let X be a classical register with the state described by a probability distribution p. The* Shannon entropy *of X is defined as:*

$$H(X) = -\sum_{x \in X} p(x) \log(p(x)). \tag{1.39}$$

Although not technically necessary, the logarithm log *is usually taken in base two, so that the Shannon entropy is measured in* bits*. When context permits, the Shannon entropy can alternatively be defined directly on a probability distribution p.*

In the case that a probability distribution has just two outcomes with probabilities λ and $1-\lambda$, the Shannon entropy reduces to the *binary entropy*:

$$h_2(\lambda) = -\lambda \log(\lambda) - (1-\lambda) \log(1-\lambda). \tag{1.40}$$

Note that, as is customary, the binary entropy is not defined in terms of a register or a probability distribution, but in terms of the parameter λ.

Probability distributions may not be independent, so that the randomness of a distribution p may be affected by the outcome of another distribution q. If X and Y are registers with states described by p and q, the *conditional entropy* $H(Y|X)$ quantifies the reduction in entropy of Y given access to the contents of the register X.

Definition 3. *For registers Y and X, the* conditional Shannon entropy *$H(Y|X)$ of Y w.r.t. X is defined as:*

$$H(Y|X) = H(X,Y) - H(X). \tag{1.41}$$

In this context, X is called the *side information*, because it can imply information regarding the register Y. It always holds that $H(Y|X) \leqslant H(Y)$, and whenever the inequality is strict, X and Y are said to be *correlated*.

Quantum entropies

Because quantum states can be mixed, there are notions of entropy associated with quantum systems as well. The most fundamental quantum entropy is a generalization of the Shannon entropy towards quantum systems, the *Von Neumann entropy*, named after the famous polymath.

Definition 4. *For a quantum state ρ, the* Von Neumann entropy $S_{\mathrm{N}}(\rho)$ *is defined as:*

$$S_{\mathrm{N}}(\rho) = -\operatorname{tr}\left[\rho \ln(\rho)\right]. \tag{1.42}$$

Using the spectral decomposition $\rho = \sum_i \lambda_i |\psi_i\rangle\langle\psi_i|$, calculating the Von Neumann entropy reduces to calculating the Shannon entropy:

$$\begin{aligned} S_{\mathrm{N}}(\rho) &= -\operatorname{tr}\left[\sum_i \lambda_i |\psi_i\rangle\langle\psi_i| \ln\left(\sum_j \lambda_j |\psi_j\rangle\langle\psi_j|\right)\right] \\ &= -\sum_{i,j} \lambda_i \ln(\lambda_j) \operatorname{tr}\left[\, |\psi_i\rangle\langle\psi_i|\, |\psi_j\rangle\langle\psi_j| \,\right] \\ &= -\sum_i \lambda_i \ln(\lambda_i) = H(p), \end{aligned} \tag{1.43}$$

where p is the probability distribution generated by the spectrum $\lambda = (\lambda_1 \ldots \lambda_n)$ of ρ, and the second equality follows from the linearity of the trace.

Like for its classical counterpart, there exists a *conditional von Neumann entropy*:

$$S_{\mathrm{N}}(\sigma|\rho) = S_{\mathrm{N}}(\rho, \sigma) - S_{\mathrm{N}}(\rho). \tag{1.44}$$

The conditional Von Neumann entropy plays a central role in many topics in quantum information science, especially in the study of entanglement, and in quantum cryptography.

An important generalisation of the Von Neumann entropy is used in part III, the so-called *conditional min entropy* $H_{\min}(A|B)$. Its general definition is somewhat involved [78, 79], but for the purposes of this thesis it can be simplified. Specifically, for a bi-partite state ρ_{XB} where the first system X is classical but B may be quantum, the conditional min entropy reduces to [1]:

$$H_{\min}(X|B) = -\log p_{\mathrm{guess}}(X|B). \tag{1.45}$$

The *conditional guessing probability* $p_{\mathrm{guess}}(X|B)$ captures how well one can guess the contents of the classical register X, given access to the quantum

system B. Any generalized measurement is allowed to be performed on B:

$$p_{\text{guess}}(X|B) = \sup_{E_x} \sum_{x \in X} \Pr[X = x] \operatorname{tr}\left[E_x \rho_{B|X=x} E_x^\dagger\right], \tag{1.46}$$

where $\rho_{B|X=x} = (\langle x| \otimes \mathbb{I}_B)\, \rho_{XB}\, (|x\rangle \otimes \mathbb{I}_B)$ is the state of the system B conditioned that the state of the classical register X is x. The E_x's form a POVM (see sec. 1.3), and the supremum is taken over all generalized measurements on the quantum system B.

Another useful entropic measure is the *conditional max entropy* $H_{\max}(A|B)$, which can be defined directly in terms of the conditional min entropy. For a state ρ_{AB} with a purification $\rho_{ABC} = |\psi\rangle\langle\psi|_{ABC}$, it is defined as [79]:

$$H_{\max}(A|B) = -H_{\min}(A|C). \tag{1.47}$$

The conditional min- and max entropies together play an important role in the security of QKD, which will be addressed in part III.

Finally, there exist *smoothed* versions of the quantum entropies, which allow for small variations in the state to be considered and make the entropies continuous. For any $\varepsilon \geqslant 0$, the *smooth conditional Von Neumann entropy* is defined as [80]:

$$S_{\text{N}}(\rho|\sigma)^\varepsilon = \sup_{\rho'} S_{\text{N}}(\rho'|\sigma), \tag{1.48}$$

where the supremum is taken over all quantum states ρ' that are at most ε-close to ρ in the *purified distance* [81]. The purified distance is a generalisation of the trace distance for *sub-normalised states* [1], which are (unphysical) states for which $\operatorname{tr}[\rho] \leqslant 1$. Nevertheless, these sub-normalized states give some operational advantages in security proofs, so that the definition of the smooth entropies is adapted to not use the standard trace distance. Note that for $\varepsilon = 0$ the non-smoothed Von Neumann entropy is retrieved. The conditional smooth min-entropy and smooth max-entropy are defined in a similar fashion.

1.5 | Entangled states

Any state $|\psi\rangle_{AB}$ defined on two quantum systems A and B can be written in a standard form called the *Schmidt decomposition* [7].

Definition 5. *Let* $|\psi\rangle_{AB} \in \mathcal{H}_A \otimes \mathcal{H}_B$ *be an arbitrary state, and let* $n = \dim(\mathcal{H}_A)$ *and* $m = \dim(\mathcal{H}_B)$. *The* Schmidt decomposition *of* $|\psi\rangle_{AB}$ *is defined as:*

$$|\psi\rangle_{AB} = \sum_{i=1}^{r} \sqrt{\lambda_i}\, |a_i\rangle \otimes |b_i\rangle, \tag{1.49}$$

where $r \leqslant \min(n, m)$ *is called the* Schmidt rank *of the state and the* λ_i*'s are called the* Schmidt coefficients*, which are subjected to the normalization*

condition $\sum_{i=1}^{r} \lambda_i = 1$. $\{|a_i\rangle\}$ *and* $\{|b_i\rangle\}$ *are sets of orthonormal states in* $\mathcal{H}_A$ *and* $\mathcal{H}_B$*, respectively, and are called the* Schmidt vectors.

A pure quantum state $|\psi\rangle_{AB}$ is called *separable* over the bipartition $A : B$ of its qubits if $|\psi\rangle_{AB} = |\psi\rangle_A \otimes |\psi\rangle_B$ for some quantum states $|\psi\rangle_A$ and $|\psi\rangle_B$, which is true if and only if its Schmidt rank $r = 1$. When its Schmidt rank r is at least 2, the state is called *entangled*, and when r is maximum (i.e. $r = \min(n, m)$, see Def. 5) it is called *maximally entangled*. For a pure state $|\psi\rangle_{AB}$, the reduced state ρ_A is mixed if and only if the state $|\psi\rangle_{AB}$ is entangled.

There exists a rich theory of entanglement; it is one of the defining properties of quantum information science, and an indispensable resource in quantum computation and communication. For a comprehensive review see [71] or [35].

Some states are more entangled than others, and quantifying entanglement is performed using *entanglement measures* [35]. For pure states, the best known entanglement measure is the *entanglement entropy* [35, 36].

Definition 6. *Let* $|\psi\rangle_{AB} \in \mathcal{H}_A \otimes \mathcal{H}_B$ *be an arbitrary state with Schmidt coefficients* $p = (\lambda_1, \lambda_2, \ldots, \lambda_r)$ *and Schmidt rank* r*. Furthermore, let* ρ_A *and* ρ_B *be the reduced states of* $|\psi\rangle_{AB}$ *on* A *and* B*, respectively.*

The entanglement entropy $\mathcal{E}_{A:B}(|\psi\rangle)$ *of* $|\psi\rangle_{AB}$ *with respect to the bipartition* $A : B$ *is defined as:*

$$\mathcal{E}_{A:B}(|\psi\rangle) = S_{\mathrm{N}}(\rho_A) = S_{\mathrm{N}}(\rho_B) = H(p). \tag{1.50}$$

From Def. 5 it follows that $S_{\mathrm{N}}(\rho_A) = S_{\mathrm{N}}(\rho_B)$*. From the same definition it follows that* p *can be viewed as a probability distribution, so that* $H(p)$ *is well-defined.*

It follows that for any state $|\psi\rangle_{AB}$ it holds that $0 \leqslant \mathcal{E}_{A:B}(|\psi\rangle) \leqslant \log(r)$, where r is the Schmidt rank of $|\psi\rangle$. Moreover, it follows that the entanglement entropy of a state is maximized exactly if the state is maximally entangled.

Quantum correlations

Entangled states can produce correlations that can not be reproduced by classical systems [36]; this is roughly known as the *EPR*-paradox [82], named after the physicists Einstein, Podolsky and Rosen who addressed it in the early stages of quantum physics.

The difference between quantum- and *classical* correlations was made more precise by Bell in his seminal work [34], where he additionally proposed a method to operationally distinguish these *quantum-* or *non-classical* from correlations allowed by classical mechanics (e.g. through *hidden variable models*). Distinguishing is usually phrased in terms of a *non-local game*, the first of which is the well-known *CHSH* game, named after the physicists

that introduced it in [83]. In such a game, two or more parties can win with a strictly higher probability by utilizing quantum states, compared to them using a purely classical strategy. Other important non-local games are the *Mermin-Peres magic square game* [84, 85] and the *GHZ-game*. Colloquially, an experiment or test that can distinguish quantum- from classical behaviour is called a *Bell test*. For a comprehensive review see [86].

The fact that only quantum systems can show these quantum correlations, can function as a test of 'quantumness'. This is the basis of some facets of quantum cryptography, which will be discussed in more detail in part III.

Examples of entangled states

The quintessential example of an entangled state is $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$, which is fundamental and ubiquitous in quantum communication. It is known as the *EPR* pair, named after the EPR-paradox and the three associated physicists [82], but the actual state itself was only popularized by Bell [34]. It can be prepared from two qubits initialized in the $|00\rangle$ state by applying a Hadamard operation on the first qubit, followed by a controlled-X operation from the first to the second qubit:

$$\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) = C_X^{1\to 2}(H \otimes \mathbb{I})\,|00\rangle\,. \tag{1.51}$$

The EPR pair is the first of the four *Bell* states, which are all maximally entangled:

$$\begin{aligned} |B_{00}\rangle &= \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)\,, \qquad |B_{01}\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle)\,, \\ |B_{10}\rangle &= \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)\,, \qquad |B_{11}\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)\,. \end{aligned} \tag{1.52}$$

For this reason, the EPR pair is often also referred to as the *Bell pair*. The four Bell states are all related by a single unitary operation on either of their qubits:

$$|B_{b_1 b_2}\rangle = (X^{b_1} Z^{b_2} \otimes \mathbb{I})\,|B_{00}\rangle = (\mathbb{I} \otimes X^{b_1} Z^{b_2})\,|B_{00}\rangle\,. \tag{1.53}$$

Sometimes the Bell states are written as $|\Phi^+\rangle, |\Phi^-\rangle, |\Psi^+\rangle$ and $|\Psi^-\rangle$ for the states $|B_{00}\rangle, |B_{01}\rangle, |B_{10}\rangle$ and $|B_{11}\rangle$, respectively. However, in this thesis mainly the latter notation will be used. Together they form the *Bell basis*.

Multi-partite entanglement

For states of three or more qubits, the *bi-partition* over which the entanglement is to be understood must be specified. Consider, for instance, the three-qubit state $|\psi\rangle_{ABC} = |B_{00}\rangle_{AB} \otimes |0\rangle_C$: it is entangled over the bi-partition $A : BC$, but separable over the partition $AB : C$.

When a state of more than two quantum systems is considered, the state can show *multi-partite entanglement*. This is an extension of the bi-partite form of entanglement, where quantum states are divided into more than two partitions.

Definition 7. *Let $|\psi\rangle_{1,2,\ldots,n}$ be an n-qubit state, and let $M \in [n]$ be an arbitrary subset of the qubits. The state $|\psi\rangle_{1,2\ldots,n}$ is* separable *over the bi-partition $M : M^{\perp}$ if it can be written as:*

$$|\psi\rangle_{1,2,\ldots,n} = |a\rangle_M \otimes |b\rangle_{M^{\perp}}, \tag{1.54}$$

where $|a\rangle_M$ and $|b\rangle_{M^{\perp}}$ are two arbitrary pure states on the qubits in M and $M^{\perp}$, respectively.

If no such bi-partition $M : M^{\perp}$ exists, the state is called genuine multi-partite entangled*, or just* multi-partite entangled*.*

In general, because the choice of M introduces arbitrary permutations of the nodes, it can be non-trivial to determine if a state is multi-partite entangled. Multi-partite entanglement will be discussed in more detail in part II.

An interesting special class of multipartite entangled states are the *absolutely maximally entangled* states [87]. These states are not just entangled over every bi-partition $M : M^{\perp}$, but are maximally entangled (see the start of this section) over every possible bi-partition.

1.6 The Clifford group and local unitary operations

The *Clifford group* is a subgroup of the n-qubit unitary group that is closely related to the Pauli operators. It plays a central role in stabilizer theory (see chapter 2), and by extension in entanglement theory [35, 71] and quantum error correction [74]. The Clifford group can be defined as the normalizer of the Pauli group in the unitary group.

Definition 8. *The n-qubit* Clifford group $\mathcal{C}_n \subset \mathcal{U}_n$ *is the normalizer $\mathcal{N}$ of the n-qubit Pauli group $\mathcal{P}_n$ in the unitary group $\mathcal{U}_n$:*

$$\mathcal{C}_n = \mathcal{N}(\mathcal{P}_n) = \{U \in \mathcal{U}_n | U\mathcal{P}_n U^{\dagger} = \mathcal{P}_n\}. \tag{1.55}$$

When an operator is in the Clifford group $\mathcal{C}_n$, it is called a Clifford operator *or just* Clifford*.*

Because the normalizer of any subgroup is itself a subgroup, it follows that the Clifford group $\mathcal{C}_n$ is a subgroup of $\mathcal{U}_n$.

Def. 8 gives an infinite subgroup, because the center $\{\alpha\mathbb{I}\}_{\alpha\in\mathbb{C}}$ of $\mathcal{U}_n$ is infinite and contained in the normalizer. However, this center represents global

phases and is thus physically irrelevant, so that it can be removed from the Clifford group without affecting its action on quantum states. This leads to a redefinition[4] of the Clifford group as a *projective group*, which is finite:

$$\mathcal{C}_n \to \mathcal{C}_n \setminus \{\alpha\mathbb{I}\}. \tag{1.56}$$

In general, in this thesis the projective group is implied, unless explicitly stated otherwise.

Some important single-qubit and two-qubit gates are Clifford, including $\mathbb{I}$ and the Pauli operators. Notably, all operators in **Tab.** 1.1 are Clifford, except the T and $P(\phi)$ gates (for general ϕ).

From Def. 8 it is straightforward that, for any $n > m$, any operator $C \in \mathcal{C}_m$ is part of $\mathcal{C}_n$, when it is suitably extended by $\mathbb{I}$ operators to be an element of $\mathcal{U}_n$.

The fact that, up to phases, the operators X_i and Z_i are generators of the n-qubit Pauli group $\mathcal{P}_n$, can be used to provide an easy test to determine if an operator $U \in \mathcal{U}_n$ is Clifford. Indeed, U is Clifford if and only if, for every $i \in [n]$, $UX_iU^\dagger \in \mathcal{P}_n$ and $UZ_iU^\dagger \in \mathcal{P}_n$.

As an example, with the use of (1.34), a straightforward computation reveals that:

$$C_Z X_1 C_Z^\dagger = C_Z(X \otimes \mathbb{I})C_Z^\dagger = (X \otimes Z) \in \mathcal{P}_n, \tag{1.57}$$

$$C_Z Z_1 C_Z^\dagger = C_Z(Z \otimes \mathbb{I})C_Z^\dagger = (Z \otimes \mathbb{I}) \in \mathcal{P}_n. \tag{1.58}$$

The C_Z gate is symmetric, so that its action on X_2 and Z_2 follows similarly. It can be concluded that C_Z is Clifford.

Local operations

A general unitary operator $U \in \mathcal{U}_n$ can not be represented as a tensor product of single-qubit operators. For example, there do not exist single-qubit unitary operators U_1 and U_2 such that $C_Z = U_1 \otimes U_2$ (see (1.34)). Those unitary operators that *can* be decomposed into single-qubit unitary operators can be interpreted as a series of single-qubit operators chained together. More specifically, if $U = U_1 \otimes U_2$, it is exactly the same as the product of $U_1 \otimes \mathbb{I}$ and $\mathbb{I} \otimes U_2$. In a sense, these operators act only on every individual qubit separately. Such *local operations* are important in a networked setting, where quantum systems might be macroscopically removed from each other, so that multi-qubit gates are hard to perform. The *local unitary group* is the set of all unitary operations that are local.

[4] Instead of defining $\mathcal{C}_n$ as the projective group $\mathcal{N}(\mathcal{P}_n) \setminus \{\alpha\mathbb{I}\}_{\alpha\in\mathbb{C}}$, one can alternatively define $\mathcal{C}_n$ in terms of a set of generators with the desired property. Usually this is taken $\langle H_i, \sqrt{Z_i}, C_X^{i\to j}\rangle$, and while its center is not trivial, this group is at least finite. See [88] for more details.

Definition 9. *The n-qubit* local unitary group $\mathcal{L}_n^{\mathcal{U}}$ *is the collection of all operators that are n-fold tensor products of single-qubit unitary operators:*

$$\mathcal{L}_n^{\mathcal{U}} = \left\{ \bigotimes_{i\in[n]} U_i | U_i \in \mathcal{U}_1 \right\}. \tag{1.59}$$

It is straightforward to show that the local unitary group is a subgroup of the unitary group $\mathcal{U}_n$*. Any operator* $U \in \mathcal{L}_n^{\mathcal{U}}$ *is called a* local unitary operator *or just* local unitary*.*

Finally, the local operators that are Clifford form the *local Clifford group.*

Definition 10. *The n-qubit* local Clifford group $\mathcal{L}_n^{\mathcal{C}}$ *is the collection of all operators that are n-fold tensor products of single-qubit Clifford operators:*

$$\mathcal{L}_n^{\mathcal{C}} = \left\{ \bigotimes_{i\in[n]} C_i | C_i \in \mathcal{C}_1 \right\}. \tag{1.60}$$

Equivalently, it is the intersection of the local unitary group and the Clifford group:

$$\mathcal{L}_n^{\mathcal{C}} = \mathcal{C}_n \cup \mathcal{L}_n^{\mathcal{U}}. \tag{1.61}$$

Any operator $C \in \mathcal{L}_n^{\mathcal{C}}$ *is called a* local Clifford operator *or just* local Clifford*.*

It is straightforward to show that the local Clifford group $\mathcal{L}_n^{\mathcal{C}}$ is a subgroup of both the Clifford group $\mathcal{C}_n$, and of the local unitary group $\mathcal{L}_n^{\mathcal{U}}$.

2

The stabilizer formalism

One defining feature of quantum states is the *superposition*, which allows them to show behaviour not possible in classical physics. However, it renders them harder to represent as well: for a general quantum state of n qubits, the number of coefficients needed to specify the state grows exponentially in n. Certain classes of states allow for more efficient representation, e.g. separable states, states with bound Schmidt number [7], bound matrix product states [89], and most notably *stabilizer states*.

Stabilizer states are the class of quantum states that can be described using the *stabilizer formalism*, which was originally developed in [73] for quantum error correction. Not all quantum states are stabilizer states, but many useful or interesting types of quantum states fall within the class. Importantly, the stabilizer formalism can represent many forms of entanglement, and allows efficient simulation of a certain class of evolutions and measurements of stabilizer states, known as *Clifford circuits*.

This chapter first introduces the stabilizer formalism in sec. 2.1, in which stabilizer states are defined as well. Unitary evolutions of stabilizer states are discussed in sec. 2.2, and measurements on stabilizer states are discussed in sec. 2.3. Section 2.4 addresses the properties of marginals of stabilizer states, which is used extensively in part II, most notably chapter 6. Finally, sec. 2.5 concludes the chapter, and gives some details about both the aforementioned Clifford circuits and possible extensions of the stabilizer formalism.

The reader familiar with the stabilizer formalism may feel free to skip this chapter, although the concepts introduced in sec. 2.4 are not necessarily part

of the standard basic introduction of the stabilizer formalism.

2.1 Stabilizer states

For a given quantum state $|\psi\rangle$, an operator O *stabilizes* $|\psi\rangle$ if $O|\psi\rangle = (+1)|\psi\rangle$, i.e. $|\psi\rangle$ is a $(+1)$-eigenstate of O.

Definition 11. *The **stabilizer** $\mathcal{S}$ of an n-qubit state $|\psi\rangle$ is the (possibly empty) collection of all Pauli operators that stabilize the state:*

$$\mathcal{S} = \{P \in \mathcal{P}_n | P|\psi\rangle = |\psi\rangle\}. \tag{2.1}$$

*A state $|\psi\rangle$ with stabilizer $\mathcal{S}$ is a **stabilizer state** if it is the unique state for which $\mathcal{S}$ is the stabilizer. If it is useful to do so, one can write $\mathcal{S}^{|\psi\rangle}$ to refer to the stabilizer of a specific state $|\psi\rangle$.*

Stabilizer states have many useful and interesting properties; most of these follow immediately from their definition. More specifically, let $|\psi\rangle$ be a stabilizer state with stabilizer $\mathcal{S}$. The product PQ of two random elements $P, Q \in \mathcal{S}$ is then necessarily in the stabilizer:

$$(PQ)|\psi\rangle = PQ|\psi\rangle = P|\psi\rangle = |\psi\rangle, \tag{2.2}$$

In particular, this means that $\mathcal{S}$ is closed. Additionally, $\mathbb{I} \in \mathcal{S}$ for any stabilizer, so that it contains an identity element. Moreover, P and Q commute on the stabilizer state:

$$PQ|\psi\rangle = P|\psi\rangle = |\psi\rangle = Q|\psi\rangle = QP|\psi\rangle. \tag{2.3}$$

Since Pauli operators either commute or anti-commute (see (1.5)), this implies that P and Q commute everywhere. It can be concluded that the stabilizer forms an Abelian subgroup of $\mathcal{P}_n$.

Let $\{g_i\}_{i=1}^{l} \subset \mathcal{S}$ be a (minimal) set of generators for $\mathcal{S}$, for some number l to be specified later. The Pauli operators are self-inverse, so due to the Abelian structure of $\mathcal{S}$, any element $P \in \mathcal{S}$ can be uniquely represent as:

$$P = g_1^{b_1} g_2^{b_2} \dots g_l^{b_l}, \tag{2.4}$$

where $b_i \in \{0, 1\}$ encodes the 'usage' of generator g_i w.r.t. P. Since there are 2^l choices of such bit strings $b = (b_1, b_2, \dots, b_l)$, it holds that $|\mathcal{S}| = 2^l$.

Moreover, when a Pauli operator $Q \in \mathcal{P}$ commutes with all the generators of a stabilizer $\mathcal{S}$, (2.4) implies that $[P, Q] = 0$ for *every* element $P \in \mathcal{S}$. This can be used to prove that any such Q has to be an element of the stabilizer $\mathcal{S}$ itself. More specifically, let $|\psi'\rangle = Q|\psi\rangle$. For every operator $P \in \mathcal{P}_n$, it holds that

$$P|\psi'\rangle = PQ|\psi\rangle = QP|\psi\rangle = Q|\psi\rangle = |\psi'\rangle, \tag{2.5}$$

i.e. $|\psi'\rangle$ is a $+1$ eigenstate for every operator $P \in \mathcal{S}$. By definition, the stabilizer state $|\psi\rangle$ is the unique state for which this holds, so that $|\psi'\rangle = Q|\psi\rangle = |\psi\rangle$. But then Q stabilizes $|\psi\rangle$, so that it is in the stabilizer $\mathcal{S}$.

It follows that $\pm Q \in \mathcal{S}$ if and only if it commutes with a generating set $\{g_i\}$ of $\mathcal{S}$:

$$Qg_i = g_i Q \quad \forall i \in \{1, 2, \ldots, n\}. \tag{2.6}$$

This gives an easy test to determine if a given element $Q \in \mathcal{P}_n$ is in the stabilizer $\mathcal{S}$ or not.

The stabilizer state is, by definition, the unique state in the shared $+1$ eigenspace of all elements of $\mathcal{S}$. Let $\Pi_{\mathcal{S}} = \prod_{P\in\mathcal{S}} \Pi^P_{+1}$ be the projector of this eigenspace, where $\Pi^P_{+1} = \frac{\mathbb{I}+P}{2}$ is the $+1$-eigenspace projector of P (see (1.10)). Using the insights of (1.12) and (2.4), the stabilizer state can then be written as:

$$|\psi\rangle\langle\psi| = \Pi_{\mathcal{S}} = \prod_{P\in\mathcal{S}} \Pi^P_{+1} = \prod_i \Pi^{g_i}_{+1} = \frac{1}{2^l}\prod_i (\mathbb{I} + g_i), \tag{2.7}$$

where l is the number of generators of $\mathcal{S}$. This shows an important fact: any stabilizer state $|\psi\rangle$ is the unique $+1$ eigenstate of merely the generators of its stabilizer instead of all 2^l elements, and therefore one can uniquely specify a stabilizer state by just a set of generators. However, note that the choice of generators for a stabilizer $\mathcal{S}$ is *not* unique. Selecting a suitable set of generators is often an important part of the analysis of a stabilizer state.

Eq. (2.7) can additionally be used to specify the number l of generators for $\mathcal{S}$. Because $|\psi\rangle$ is the unique state in the shared $+1$-eigenspaces of all Pauli operators, the dimension of the subspace associated with the projector $\Pi_{\mathcal{S}}$ is 1. The dimension of a subspace is equal to the trace of its projector, so

$$1 = \operatorname{tr}[\Pi_{\mathcal{S}}] = \operatorname{tr}\left[\prod_{P\in\mathcal{S}} \Pi^P_{+1}\right] = \frac{\operatorname{tr}[\mathbb{I}]}{2^l} = 2^{n-l}, \tag{2.8}$$

where it is used that all Pauli operators except the $\mathbb{I}$ operator are traceless (see (1.7)). This means that the number l of generators of $\mathcal{S}$ equals n and therefore that there are exactly n generators needed to specify an n-qubit stabilizer state. If there are fewer generators, the stabilized subspace (i.e. the subspace associated with the projector $\prod_{P\in\mathcal{S}} \Pi^P_{+1}$) is of dimension 2^{n-l}. In this case (2.7) can still be used to describe the (mixed) state, except that it would not be properly normalized. Such mixed states are discussed in more detail in sec. 2.4.

Combining (2.7) with the insights from (2.4) allows the stabilizer state $|\psi\rangle\langle\psi|$ to be represented as a sum of all stabilizer elements:

$$|\psi\rangle\langle\psi| = \frac{1}{2^n}\prod_i (\mathbb{I} + g_i) = \frac{1}{2^n}\sum_{b\in\{0,1\}^n} g_1^{b_1} g_2^{b_2} \ldots g_n^{b_n} = \frac{1}{2^n}\sum_{P\in\mathcal{S}} P. \tag{2.9}$$

From (2.9), it is evident that there is a one-to-one correspondence between stabilizer states and their stabilizer. Every stabilizer state uniquely determines its stabilizer, and every stabilizer uniquely determines its associated stabilizer state: the only freedom in describing a stabilizer state is the choice of generators.

Additionally, it follows that *any* set $\{P_i\}_{i=1}^n$ of n Pauli operators that both pairwise commute and are independent (i.e. they are not a product of each other), generates a valid stabilizer $\mathcal{S}$ and associated stabilizer state.

2.1.1 | Fidelity of arbitrary states with stabilizer states

The fidelity of an arbitrary state ρ with a stabilizer state can be computed using (2.9). Because the stabilizer state is pure, (1.28) can be used and

$$F(\rho, |\psi\rangle) = \mathrm{tr}\left[\rho\,|\psi\rangle\langle\psi|\right] = \mathrm{tr}\left[\rho\left(\frac{1}{2^n}\sum_{P\in\mathcal{S}} P\right)\right] = \frac{1}{2^n}\sum_{P\in\mathcal{S}} \mathrm{tr}\left[\rho P\right], \tag{2.10}$$

where the last equality follows from the linearity of the trace. The terms $\mathrm{tr}\left[\rho P\right]$ are expectation values of simple Pauli-basis measurements; this plays an important role in the verification of stabilizer states, which will be addressed in part III.

2.1.2 | Examples of stabilizer states

One of the most straightforward examples of a stabilizer state is the state $|+\rangle$. $X|+\rangle = |+\rangle$, so its stabilizer is $\mathcal{S}^{|+\rangle} = \{\mathbb{I}, X\}$, which is generated by a single generator $g_1 = X$. Similarly, the state $|1\rangle$ is the $+1$ eigenstate of the operator $-Z$, so its stabilizer is $\mathcal{S}^{|1\rangle} = \{\mathbb{I}, -Z\}$, generated by a single generator $g_1 = -Z$.

A less trivial examples is that of the Bell states (see (1.52)). A straightforward computation reveals that $(X\otimes X)|B_{00}\rangle = |B_{00}\rangle$ and $(Z\otimes Z)|B_{00}\rangle = |B_{00}\rangle$. Additionally, these operators commute: $[X\otimes X, Z\otimes Z] = 0$. It follows that $|B_{00}\rangle$ is a stabilizer state with generators $g_1 = X\otimes X$ and $g_2 = Z\otimes Z$, and with the stabilizer $\mathcal{S}^{|B_{00}\rangle} = \{\mathbb{I}, XX, ZZ, -YY\}$, where the '$\otimes$'-sign is dropped for brevity. The other three Bell states follow similarly, as listed in **Tab.** 2.1.

2.2 | Unitary evolutions of stabilizer states

When a state $|\psi\rangle$ is rotated under a unitary transformation U, its density matrix $\rho = |\psi\rangle\langle\psi|$ is evolved as $\rho \to U\rho U^\dagger$. The unitary evolution of a

	g_1	g_2	$\mathcal{S}$
$\|B_{00}\rangle$	$X \otimes X$	$Z \otimes Z$	$\{\mathbb{I}, \quad XX, \quad ZZ, -YY\}$
$\|B_{01}\rangle$	$X \otimes X$	$-Z \otimes Z$	$\{\mathbb{I}, \quad XX, -ZZ, \quad YY\}$
$\|B_{10}\rangle$	$-X \otimes X$	$Z \otimes Z$	$\{\mathbb{I}, -XX, \quad ZZ, \quad YY\}$
$\|B_{11}\rangle$	$-X \otimes X$	$-Z \otimes Z$	$\{\mathbb{I}, -XX, -ZZ, -YY\}$

Table 2.1: The four Bell states (see (1.52)) are stabilizer states that are closely related to each other. For the Bell state $|B_{b_1 b_2}\rangle = (Z^{b_1} X^{b_2} \otimes \mathbb{I}) \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle)$, its stabilizer is generated by $g_1 = (-1)^{b_2} XX$ and $g_2 = (-1)^{b_1} ZZ$, resulting in the stabilizer $\mathcal{S}^{|B_{b_1 b_2}\rangle}$ with four elements.

stabilizer state $|\psi\rangle\langle\psi|$ with generators $\{g_i\}$ can be calculated using (2.7):

$$\begin{aligned} |\psi\rangle\langle\psi| = \frac{1}{2^n}\prod_i (\mathbb{I} + g_i) \to \; & U\left(\frac{1}{2^n}\prod_i (\mathbb{I} + g_i)\right) U^\dagger \\ =& \frac{1}{2^n} U\Big((\mathbb{I} + g_1)(\mathbb{I} + g_2)\ldots(\mathbb{I} + g_n)\Big) U^\dagger \\ =& \frac{1}{2^n} U\Big((\mathbb{I} + g_1)U^\dagger U(\mathbb{I} + g_2)U^\dagger U \ldots U^\dagger U(\mathbb{I} + g_n)\Big) U^\dagger \\ =& \frac{1}{2^n}\prod_i (\mathbb{I} + U g_i U^\dagger), \end{aligned} \tag{2.11}$$

where $U^\dagger U = \mathbb{I}$ is freely introduced in the third row, and where the last equality uses $U(\mathbb{I} + g_i)U^\dagger = \mathbb{I} + U g_i U^\dagger$. This means that the rotated state is the shared $+1$ eigenspace of the elements of the group $\langle U g_i U^\dagger \rangle$. However, the operators $U g_i U^\dagger$ may not be Pauli operators, so that the state may fail to be a stabilizer state.

This representation is especially useful when $U \in \mathcal{C}$ is a Clifford operator, because then $U g_i U^\dagger$ is (guaranteed to be) in $\mathcal{P}$. Moreover, conjugation with a unitary operator preserves commutation relations, and the rotated generators thus correctly create a stabilizer group. Therefore, the rotated stabilizer state is a stabilizer state as well, with associated generators:

$$g_i \to g_i' = C g_i C^\dagger. \tag{2.12}$$

2.2.1 | Examples of evolutions of stabilizer states

As an example, (the generators of) the Bell pair $|B_{00}\rangle$ can be computed directly with (2.12). The state $|00\rangle$ can be used to prepare the Bell pair $|B_{00}\rangle$ by applying a Hadamard operation to the first qubit, followed by a $C_X^{1\to 2}$ gate controlled by that same qubit. Therefore, the generators of the Bell pair can be obtained by evaluating the action of these unitary operators on the generators of the $|00\rangle$ state. The state $|00\rangle$ has generators $g_1 = Z \otimes \mathbb{I}$ and

$g_2 = \mathbb{I} \otimes Z$, so (using (1.51)) these generators are first transformed by $H \otimes \mathbb{I}$:

$$\begin{array}{rcrcl} Z \otimes \mathbb{I} & \to & (H \otimes \mathbb{I})(Z \otimes \mathbb{I})(H \otimes \mathbb{I})^\dagger & = & X \otimes \mathbb{I}, \\ \mathbb{I} \otimes Z & \to & (H \otimes \mathbb{I})(\mathbb{I} \otimes Z)(H \otimes \mathbb{I})^\dagger & = & \mathbb{I} \otimes Z, \end{array} \tag{2.13}$$

and subsequently by $C_X^{1\to 2}$:

$$\begin{array}{rcrcl} X \otimes \mathbb{I} & \to & C_X^{1\to 2}(X \otimes \mathbb{I})(C_X^{1\to 2})^\dagger & = & X \otimes X, \\ \mathbb{I} \otimes Z & \to & C_X^{1\to 2}(\mathbb{I} \otimes Z)(C_X^{1\to 2})^\dagger & = & Z \otimes Z. \end{array} \tag{2.14}$$

The generators of the other Bell states follow readily. E.g. for the Bell state $|B_{10}\rangle = (X \otimes \mathbb{I})\,|B_{00}\rangle$ (see (1.53)), the discussion before (2.12) implies that its generators are:

$$\begin{array}{rcrcl} X \otimes X & \to & (X \otimes \mathbb{I})(X \otimes X)(X \otimes \mathbb{I})^\dagger & = & X \otimes X, \\ Z \otimes Z & \to & (X \otimes \mathbb{I})(Z \otimes Z)(X \otimes \mathbb{I})^\dagger & = & -Z \otimes Z. \end{array} \tag{2.15}$$

Note that these generators coincide with the generators as listed in **Tab.** 2.1.

2.3 Measurements on stabilizer states

For any Pauli operator $O \in \mathcal{P}_n$ interpreted as an observable, it is straightforward to compute the expectation value $\mathbb{E}[O]$ for a stabilizer state $|\psi\rangle$:

$$\mathbb{E}[O] = \langle\psi|\, O\, |\psi\rangle = \operatorname{tr}[O\,|\psi\rangle\langle\psi|] = \operatorname{tr}\left[O\Big(\frac{1}{2^n}\sum_{P\in\mathcal{S}} P\Big)\right] = \frac{1}{2^n}\sum_{P\in\mathcal{S}} \operatorname{tr}[OP], \tag{2.16}$$

where the second equality follows from (2.9). This expectation value shows different behaviour depending on if O or $-O$ is in the stabilizer, or neither of them are included:

$$\mathbb{E}[O] = \begin{cases} 1 & O \in \mathcal{S}, \\ -1 & -O \in \mathcal{S}, \\ 0 & \pm O \notin \mathcal{S}. \end{cases} \tag{2.17}$$

The first two cases are straightforward. When $O \in \mathcal{S}$, the term $O^2 = \mathbb{I}$ exists in the sum in (2.16), which has trace 1 when normalized. Similarly, when $-O \in \mathcal{S}$ the term $(-O)O = -\mathbb{I}$ exists in the sum, which has trace -1 when normalized. All other elements in the stabilizer are traceless, so that the sum in (2.16) equates to $+1$ or -1, respectively.

The last case follows from the fact that when $O \notin \mathcal{S}$, the set $O\mathcal{S}$ is a (left) coset of $\mathcal{S}$ and thus does not contain $\pm\mathbb{I}$. That means that all terms in $O\mathcal{S}$ are traceless, and the sum in (2.16) equals zero.

Any observable A can be written as $A = \sum_{P\in\mathcal{P}} \alpha_P P$. By the linearity of the trace, (2.17) generalises to any other observable:

$$\mathbb{E}[A] = \sum_{P\in\mathcal{P}} \alpha_P \mathbb{E}[P] = \sum_{P\in\mathcal{S}} \alpha_P - \sum_{-P\in\mathcal{S}} \alpha_P, \tag{2.18}$$

In the case that O is a Pauli operator, the measurement will result in an outcome $m = +1$ or $m = -1$. Using the identity $\mathbb{E}[O] = \Pr(m = +1) - \Pr(m = -1)$ together with (2.17) it follows that this measurement either always has the same outcome, or that it is uniformly random:

$$\Pr(m_O = +1), \Pr(m_O = -1) = \begin{cases} 1, 0 & O \in \mathcal{S}, \\ 0, 1 & -O \in \mathcal{S}, \\ \frac{1}{2}, \frac{1}{2} & \pm O \notin \mathcal{S}. \end{cases} \tag{2.19}$$

2.3.1 | Post-measurement states of Pauli measurements

The post-measurement state of such a Pauli-measurement can be determined from its generators, and will be a stabilizer state as well. More specifically, the post-measurement state $|m\rangle$ is the projection of $|\psi\rangle$ upon the eigenspace according to the measurement outcome m (see (1.36)). The projection operator for the measurement outcome $m = \pm 1$ is $\Pi^O_{(m)} = \frac{\mathbb{I} \pm O}{2}$ (see (1.10)), so the post-measurement state is, up to a normalization factor, $|m\rangle = \Pi^O_{(m)} |\psi\rangle$.

When either O or $-O$ is in the stabilizer $\mathcal{S}$, the state $|\psi\rangle$ is already an eigenstate of O or $-O$, respectively. In these cases the stabilizer state is unaffected by the measurement (or obtains a global phase -1).

When neither O nor $-O$ is in the stabilizer $\mathcal{S}$, the stabilizer state $|\psi\rangle$ is not an eigenstate of O and will be non-trivially affected by the measurement. Nevertheless, the post-measurement state is still a stabilizer state whose generators can be determined. First, if $\pm O \notin \mathcal{S}$, by (2.5) and the analysis below it, there exists at least one generator g_i that does not commute with O (and therefore anti-commutes).

W.l.o.g. assume that only the first m generators anti-commute with O (where $1 \leq m \leq n$). It is straightforward to define a change of generators $g_i \to g'_i$ for the stabilizer $\mathcal{S}$ so that afterwards only one generator anti-commutes:

$$g_i \to g'_i = \begin{cases} g_i & i = 1, \\ g_i g_1 & 2 \leq i \leq m, \\ g_i & i > m. \end{cases} \tag{2.20}$$

By construction, only the generator g'_1 anti-commutes with O, while all other generators commute with the measurement operator.

By definition, the post-measurement state $|m\rangle$ has to be an m-valued eigenstate of O, so that $|m\rangle$ is stabilized by $(m)O$. However, using (1.11) one can show that $|m\rangle$ is still stabilized by any generator g'_i that commutes with O:

$$g'_i |m_O\rangle = g'_i \Pi^O_{(m_O)} |\psi\rangle = \Pi^O_{(m_O)} g'_i |\psi\rangle = \Pi^O_{(m_O)} |\psi\rangle = |m_O\rangle . \tag{2.21}$$

The post-measurement state $|m\rangle$ is still stabilized by the generators $\{g'_i\}_{i=2}^n$, and additionally stabilized by $(m)O$. All of these operators commute by construction, and are independent of each other. This means

that they form a valid set of n generators, so that they form a valid stabilizer $\mathcal{S}^{|m\rangle}$(see (2.9) and the discussion after it). In conclusion, the post-measurement state $|m\rangle$ is a stabilizer state with the stabilizer $\mathcal{S}^{|m\rangle} = \langle (m)O, g_2', g_3', \ldots, g_n' \rangle$.

2.3.2 | Examples of Pauli measurements

As a straightforward example of a Pauli-basis measurement, consider again the Bell state $|B_{00}\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ with generators $X \otimes X$ and $Z \otimes Z$ (see **Tab.** 2.1). When the first qubit is measured in the X-basis, i.e. the basis $\{|+\rangle, |-\rangle\}$ with outcome $+1$ or -1, the associated observable is $X \otimes \mathbb{I}$. This operator does not commute with $Z \otimes Z$ so it follows that $\pm X \otimes \mathbb{I} \notin \mathcal{S}$ (see (2.6)). It immediately follows from (2.19) that the measurement results in a uniformly random outcome $m = \pm 1$.

The only generator that the measurement operator does not commute with is $Z \otimes Z$. For the post-measurement state only this generator is replaced (i.e. there is no change of generators needed), and the post-measurement state is stabilized by $X \otimes X$ and $(m)X \otimes \mathbb{I}$.

Although technically not necessary, a change of generators is instructive: applying the second generator to the first results in $X \otimes X \to (X \otimes X)((m)X \otimes \mathbb{I}) = (m)\mathbb{I} \otimes X$. The post-measurement state after measuring $X \otimes \mathbb{I}$ is thus given by the generators $(m)X \otimes \mathbb{I}$ and $(m)\mathbb{I} \otimes X$. If the measurement outcome was $m = 0$ or $m = 1$, the post-measurement state would be $|+\rangle \otimes |+\rangle$ or $|-\rangle \otimes |-\rangle$, respectively. See **Tab.** 2.2 for a detailed analysis of the same measurement.

	meas.		**c.o.g.**	
$X \otimes X$	$\to$	$X \otimes X$	$\to$	$(m)\mathbb{I} \otimes X$
$Z \otimes Z$	$\to$	$(m)X \otimes \mathbb{I}$	$\to$	$(m)X \otimes \mathbb{I}$

Table 2.2: Two qubits are initially in the state stabilized by $X \otimes X$ and $Z \otimes Z$, which is the EPR pair $|B_{00}\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ (see **Tab.** 2.1). The first qubit is measured in the X-basis (labelled **meas.**), represented by the measurement operator $X \otimes I$. The anti-commuting generator $Z \otimes Z$ is replaced by the measurement operator $(m)X \otimes I$, that now carries the uniformly random measurement outcome $m \in \{+1, -1\}$ as a phase. A change of generators (labelled **c.o.g.**) shows that the post-measurement state has generators $(m)X \otimes \mathbb{I}$ and $(m)\mathbb{I} \otimes X$, which shows that the second qubit has 'collapsed' as well to $|+\rangle$ or $|-\rangle$ when m is $+1$ or -1, respectively.

A less trivial example is given in **Tab.** 2.3. Here, the initial state is the four-qubit state $|B_{00}\rangle \otimes |B_{00}\rangle$, which is a four-qubit stabilizer state with generators $XXII, ZZII$ and $IIXX, IIZZ$. The second and third qubit are

measured in the Bell basis (see (1.52)), which results in that these two qubits are in one of the four Bell states $|B_{m_2,m_1}\rangle$, where m_1 and m_2 specify the measurement outcome. An interesting result is shown in **Tab.** 2.3: the first and last qubits are in the state $|B_{m_2,m_1}\rangle$ as well! Those familiar with it might recognize this as *entanglement swapping*, i.e. the swapping of pairs of qubits that share entanglement. This is the basic building block of the *quantum repeater*, where entanglement between a network node and a midway station, and entanglement between the midway station and another node, is swapped for entanglement between the two nodes.

c.o.g.		**meas.** 1		**c.o.g.**		
$XXII$	$\rightarrow$	$XXII$	$\rightarrow$	$XXII$	$\rightarrow$	$XXII$
$ZZII$	$\rightarrow$	$ZZII$	$\rightarrow$	$(m)IXXI$	$\rightarrow$	$(m_1)IXXI$
$IIXX$	$\rightarrow$	$XXXX$	$\rightarrow$	$XXXX$	$\rightarrow$	$(m_1)XIIX$
$IIZZ$	$\rightarrow$	$ZZZZ$	$\rightarrow$	$ZZZZ$	$\rightarrow$	$ZZZZ$

	meas. 2		**c.o.g.**	
$XXII$	$\rightarrow$	$(m_2)IZZI$	$\rightarrow$	$(m_2)IZZI$
$(m_1)IXXI$	$\rightarrow$	$(m_1)IXXI$	$\rightarrow$	$(m_1)IXXI$
$(m_1)XIIX$	$\rightarrow$	$(m_1)XIIX$	$\rightarrow$	$(m_1)XIIX$
$ZZZZ$	$\rightarrow$	$ZZZZ$	$\rightarrow$	$(m_2)ZIIZ$

Table 2.3: Bell-state measurement on the second and third qubit of a four-partite quantum state $|\psi\rangle = |B_{00}\rangle \otimes |B_{00}\rangle$, that has generators $XXII, ZZII$ and $IIXX, IIZZ$. **(Top)** A change of generators (labelled **c.o.g.**) facilitates the measurement of the operator $IXXI$, so that this operator anti-commutes with only one generator, $ZZII$. The measurement (labelled **meas.** 1) thus replaces this operator with the measurement operator $(m_1)IXXI$. Another change of generators results in the generators listed in the top right.
(Bottom) The measurement (labelled **meas.** 2) of the operator $IZZI$ is performed; this operator anti-commutes with only the generator $XXII$. This operator is thus replaced by the measurement operator $(m_2)IZZI$. Another change of generators (labelled **c.o.g.**) results in the generators listed in the bottom right. By **Tab.** 2.1, the state of the second and third qubit is the state $|B_{m_2,m_1}\rangle$. As a consequence, the state of the other two qubits is $|B_{m_2,m_1}\rangle$ as well. The reader that is familiar with the concept might identify this as *entanglement swapping*.

2.4 | Reduced states and bipartite entanglement

The marginals of stabilizer states show structure that is useful to study the entanglement of these states, and they will play a central role in chapter 6. Calculating the marginals of stabilizer states can be done within the stabilizer formalism, and is facilitated by (2.9). More specifically, let $|\psi\rangle$ be any stabilizer state with stabilizer $\mathcal{S}$, and let $M \subset \{1, 2, \dots n\}$ be any selection of the qubits of $|\psi\rangle$ with size $|M| = k$. The goal is to compute the reduced state $\rho_M = \mathrm{tr}_{M^\perp}[|\psi\rangle\langle\psi|]$. To this end, it is useful to introduce another concept first, the *reduced stabilizer*.

Definition 12. *Let $\mathcal{S}$ be any n-qubit stabilizer, and let $M \subset \{1, 2, \dots n\}$ be a subset of size $|M| = k$. Write $P = \bigotimes_{i=1}^{n} P_i$ for every operator $P \in \mathcal{S}$.*

The reduced stabilizer *$\mathcal{S}_M \subset \mathcal{P}_k$ is then the collection of k-qubit Pauli operators $P' = \bigotimes_{i \in M} P_i$ for every $P \in \mathcal{S}$ whose support* $\mathrm{supp}(P)$ *(see (1.6)) is contained in M:*

$$\mathcal{S}_M = \left\{ P' = \bigotimes_{i \in M} P_i | P = \bigotimes_{i=1}^{n} P_i \in \mathcal{S}, \mathrm{supp}(P) \subseteq M \right\}. \tag{2.22}$$

Alternatively, it can be defined immediately from the elements of $\mathcal{S}$ by tracing away the qubits outside of M and renormalizing:

$$\mathcal{S}_M = \left\{ \frac{1}{2^{n-k}} \mathrm{tr}_{M^\perp}[P] \,|\, P \in \mathcal{S}, \mathrm{supp}(P) \subseteq M \right\}, \tag{2.23}$$

where the scaling factor $\frac{1}{2^{n-k}}$ is there to renormalize, so that indeed $\mathcal{S}_M \subset \mathcal{P}_k$. When context permits, $\mathcal{S}_M$ can additionally be referred to as the reduced stabilizer of the marginal M.

In other words, the reduced stabilizer $\mathcal{S}_M$ is the collection of all elements $P \in \mathcal{S}$ with support contained in M, when their $\mathbb{I}$'s on the qubits outside of M are removed by tracing them away, after which they are properly renormalized.

It is straightforward to show that $\mathcal{S}_M$ is an Abelian subgroup of $\mathcal{P}_k$ with a number of generators $1 \leqslant d_M \leqslant k$, where the *dimension* d_M of $\mathcal{S}_M$ is the number of elements in a (minimum) generating set for it.

Because its elements are self-inverse and $\mathcal{S}$ is an Abelian subgroup (see (2.4)), the dimension d_M is exactly the base-two logarithm of the number of elements in $\mathcal{S}_M$:

$$d_M = \log(|\mathcal{S}_M|). \tag{2.24}$$

Finally, $\mathcal{S}_M$ stabilizes a *subspace*[1] of $\mathcal{H}$ of dimension 2^{k-d_M} (see (2.8) and the discussion immediately afterwards). The number of generators d_M and number of nodes $k = |M|$ in (2.24) are the total number of qubits n and number of generators l in (2.8), respectively.

[1] For those familiar, $\mathcal{S}_M$ forms a *stabilizer code* with $k - d_M$ logical qubits, although a pretty bad one for most choices of $\mathcal{S}$ and M.

Reduced states of stabilizer states

Using the reduced stabilizer $\mathcal{S}_M$, the marginal $\rho_M = \mathrm{tr}_{M^\perp}[|\psi\rangle\langle\psi|]$ (see (1.25)) of the stabilizer state $|\psi\rangle\langle\psi| = \frac{1}{2^n}\sum_{P\in\mathcal{S}} P$ (see (2.9)) can be calculated:

$$\begin{aligned}\rho_M &= \mathrm{tr}_{M^\perp}[|\psi\rangle\langle\psi|] \\ &= \frac{1}{2^n}\sum_{P\in\mathcal{S}} \mathrm{tr}_{M^\perp}[P], \\ &= \frac{1}{2^{|M|}}\sum_{P'\in\mathcal{S}_M} P',\end{aligned} \tag{2.25}$$

where the first equality follows from the linearity of the trace, and the second equality follows from (1.8). If it is the case that $\mathcal{S}_M = \{\mathbb{I}\}$, the marginal ρ_M is the maximally mixed state, and it is called *trivial*.

Using the insights from (2.7) and (2.9), the reduced state ρ_M is exactly the maximally mixed state in the subspace stabilized by $\mathcal{S}_M$. Moreover, if $\{g_i\}_{i=1}^{d_M}$ is a generating set for $\mathcal{S}_M$, then:

$$\begin{aligned}\rho_M &= \frac{1}{2^{|M|}}\sum_{P\in\mathcal{S}_M} P \\ &= \frac{1}{2^{|M|}}\prod_{i=1}^{d_M}(\mathbb{I}+g_i) \\ &= \frac{1}{\mathrm{rnk}(\rho_M)}\prod_{i=1}^{d_M}\Pi_i \\ &= \frac{1}{\mathrm{rnk}(\rho_M)}\sum_{j=1}^{\mathrm{rnk}(\rho_M)}|\psi_j\rangle\langle\psi_j|,\end{aligned} \tag{2.26}$$

where $\mathrm{rnk}(\rho_M) = 2^{|M|-d_M}$ is the rank of ρ_M, and $|\psi_j\rangle$ forms a basis for the shared $(+1)$-eigenspace of the generators g_i of $\mathcal{S}_M$, i.e. the subspace stabilized by $\mathcal{S}_M$.

Furthermore, the rank of ρ_M equals the Schmidt rank r of $|\psi\rangle$, so using this expression the Schmidt rank of any stabilizer state $|\psi\rangle$ (w.r.t. the bipartition $M : M^\perp$) can be calculated:

$$r = \mathrm{rnk}(\rho_M) = 2^{|M|-d_M}. \tag{2.27}$$

It follows that the Schmidt rank for a stabilizer state is always a power of two.

Finally, for any choice of bipartition $M : M^\dagger$ it holds that $\mathrm{rnk}(\rho_M) = \mathrm{rnk}(\rho_{M^\perp})$. Therefore, from (2.27) it follows that:

$$|M| - d_M = \left|M^\perp\right| - d_{M^\perp}. \tag{2.28}$$

This means that $d_{M^\perp}$ is determined by d_M.

Bipartite entanglement for stabilizer states

For any n-qubit stabilizer state $|\psi\rangle$ with stabilizer $\mathcal{S}$, and any bipartition $M : M^\perp$, the entanglement entropy $\mathcal{E}_{M:M^\perp}(|\psi\rangle)$ (see Def. 6) can be calculated using (2.26):

$$\begin{aligned}\mathcal{E}_{M:M^\perp}(|\psi\rangle) &= S_{\mathrm{N}}(\rho_M) \\ &= -\sum_{j=1}^{2^{|M|-d_M}} \frac{1}{2^{|M|-d_M}} \log\left(\frac{1}{2^{|M|-d_M}}\right) \\ &= |M| - d_M,\end{aligned} \tag{2.29}$$

where $S_{\mathrm{N}}(\rho_M)$ is the Von Neumann entropy of ρ_M (see (1.42)) and where it is used implicitly that the Schmidt rank of $|\psi\rangle$ is $r = 2^{|M|-d_M}$.

When the stabilizer state $|\psi\rangle$ with stabilizer $\mathcal{S}$ is separable over the bipartition $M : M^\perp$, it holds that $|\psi\rangle = |\psi_M\rangle \otimes |\psi_{M^\perp}\rangle$, i.e. ρ_M is a pure state. In that case, this means that the subspace that $\mathcal{S}_M$ stabilizes has dimension one, and thus that $d_M = |M|$ (see (2.26)). The same applies to $\rho_{M^\perp}$, so it follows (for separable stabilizer states) that:

$$\mathcal{S} = \mathcal{S}_M \otimes \mathcal{S}_{M^\perp}. \tag{2.30}$$

This deconstruction can be generalized to stabilizer states with an arbitrary Schmidt rank r [90]. Let $\{a_j \otimes \mathbb{I}_{M^\perp}\}_{j=1}^{d_M}$ be the generators for $\mathcal{S}_M$ and $\{\mathbb{I}_M \otimes b_k\}_{k=1}^{d_{M^\perp}}$ be the generators for $\mathcal{S}_{M^\perp}$, both extended to be elements of $\mathcal{P}_n$.

This gives a total of $d_M + d_{M^\perp}$ generators; there need to be n generators in total, so there are $n - d_M - d_{M^\perp} = 2\log(r)$ other generators.

These other generators can always be chosen *canonical form* [90], which are generators that show a rich, useful structure. More explicitly, these canonical generators form $\log(r)$ pairs (g_i, h_i) (for $1 \leqslant i \leqslant \log(r)$), defined as:

$$\begin{aligned} g_i &= g_M^i \otimes g_{M^\perp}^i, \\ h_i &= h_M^i \otimes h_{M^\perp}^i. \end{aligned} \tag{2.31}$$

Although g_i commutes with any other generator of $\mathcal{S}$ due to its Abelian structure, the projections g_M^i and $g_{M^\perp}^i$ do not necessarily commute with the other (projections of) generators.

However, the canonical form dictates that the projections of the pairs (g_i, h_i) anti-commute with each other:

$$\{g_M^i, h_M^i\} = 0, \tag{2.32}$$

but both commute with all other projections (i.e. $g_M^{i'}$ and $h_M^{i'}$ for any $i' \neq i$, and every a^j). The projections $g_{M^\perp}^i$ and $h_{M^\perp}^i$ have similar commutation relations.

These pairs (g_i, h_i) together form another subgroup $\mathcal{S}_{M:M^\perp}$, so that the stabilizer $\mathcal{S}$ has the structure

$$\mathcal{S} = (\mathcal{S}_M \otimes \mathcal{S}_{M^\perp}) \cdot \mathcal{S}_{M:M^\perp}, \tag{2.33}$$

where $\cdot$ denotes the product of the two subgroups. In a sense, all entanglement properties of $|\psi\rangle$ are encoded into the group $\mathcal{S}_{M:M^\perp}$, while the local information of the subsystems M and $M^\perp$ are encoded into $\mathcal{S}_M$ and $\mathcal{S}_{M^\perp}$, respectively.

2.5 | Conclusion and further reading

The stabilizer formalism is ubiquitous in both quantum computation and communication, with usage in quantum error correction, fault tolerance, entanglement- distillation and distribution in networks. Following (2.12) and sec. 2.3, any circuit with only Clifford operators and Pauli measurements can be simulated efficiently, something which is known as the *Gottesman-Knill theorem* [91, 92]. This efficient simulation is facilitated by the *binary representation* of Pauli operators [93], which allows to represent a stabilizer as a subspace in $\mathbb{F}_4^n$, which is then usually mapped to a symplectic subspace of $\mathbb{F}_2^{2n}$ [7]. This representation will be used in chapter 4.

Even though many interesting and highly entangled quantum states can be represented within the stabilizer formalism, and thus efficiently simulated, the stabilizer formalism can only simulate the action of Clifford operators, so that $\mathbb{BQP}$-complete circuits remain intractable. It shows an intricate interplay with the theory of fault-tolerant quantum computation [75], especially because of the fact that including one more type of gate into the circuit (usually the $T = \mathrm{diag}(1, e^{i\frac{\pi}{4}})$ gate) alleviates its power to be $\mathbb{BQP}$-complete [7, 94].

To circumvent these shortcomings and allow for a larger class of states to be represented, the stabilizer formalism can be extended to include other operators than the Pauli operators. Because the Pauli Y operator can be written as the product iXZ, all stabilizer elements (in the standard framework) are elements of the group[2] $\langle -\mathbb{I}, X, Z\rangle^{\otimes n}$. The first well-known extension of the stabilizer formalism introduces stabilizer operators that are elements of the group $\langle i\mathbb{I}, X, S = \mathrm{diag}(1, e^{i\frac{\pi}{2}})\rangle^{\otimes n}$ [95], and is thus known as the XS-stabilizer formalism. Note that this means that the elements of the stabilizer then *do not* necessarily commute any more.

More recent work gives a family of stabilizer extensions for any choice of natural number N. It introduces ω as a $2N$-th root of unity and $P = \mathrm{diag}(1, \omega^2)$ as an N-th root of Z; the stabilizer elements are then elements of the group $\langle \omega\mathbb{I}, X, P\rangle^{\otimes n}$. As such, it is known as the XP-stabilizer formalism [96]. Note that $N = 1$ retrieves the standard stabilizer formalism, and

[2] Remember that a stabilizer element can only have a phase ± 1.

$N = 2$ retrieves the XS-formalism. To compare against these extensions, the standard stabilizer formalism is sometimes referred to as the XZ-formalism.

These extensions can indeed represent more states, but this comes at a reduction in efficiency of simulation. (Note that simulation of the XP-formalism for arbitrarily large N gives $\mathbb{BQP}$-completeness).

Although bipartite entanglement of stabilizer states can be characterized using the methods introduced in sec. 2.4, multi-partite entanglement is less straightforward. Both in quantum computation and in quantum networks with more than two parties, (multi-partite) entanglement is an important resource, so that the characterization of multi-partite entanglement is exceedingly useful. The study of multi-partite entanglement is helped by an important subclass of the stabilizer states: the *graph states*. These are introduced and defined in chapter 3, where additionally important basic results are stated. Making use of these new concepts, various facets of multi-partite entanglement are then addressed in part II.

3

Graph states

Although stabilizer states permit an efficient and straightforward description in terms of a generating set of their stabilizer, it is not always immediately clear how to interpret these operators. Moreover, it can be tedious to analyse the action of unitary evolutions or Pauli measurements by hand, or to determine certain interesting properties of the state (e.g. if it is separable under a certain bipartition).

A specific subset of the stabilizer states, known as the *graph states*, allows for a much quicker and more intuitive inspection and understanding. These useful properties mostly arise from the fact that they can be represented by the mathematical concept of *graphs*, or (depending on the perspective) can even be defined in terms of them. A graph, a collection of points and potential lines between them, can be easily drawn on e.g. a piece of paper, which facilitates convenient inspection. This chapter introduces all their concepts that are relevant for this thesis; a comprehensive introduction of graph states and their properties can be found in [3].

Many of the interesting properties of graph states can be seen in terms of properties of their underlying graphs, so that often it is enough to merely inspect the drawing of this graph by hand to determine the properties of the graph state. As an explicit example, there exist an intricate relation between local Clifford operations on graph states and a specific graph-theoretic operation known as *local complementation*. Additionally, the action of single-qubit Pauli measurements on graph states can be understood in terms of their underlying graphs as well, so that the effect of these measurements can be computed

from the graphs directly. This chapter introduces the necessary concepts of graphs and graph states, so that they can be used to study entanglement and other properties of both graph- and stabilizer states in part II.

Section 3.1 gives the mathematical definition of a graph and introduces some of its relevant concepts, including the local complementation. Graph states themselves are then defined in sec. 3.2, where some examples are given as well. Among these examples is the GHZ state, which is an important resource in quantum communication and will play a central role in both chapter 5 and part III. In sec. 3.3 the relation between the local complementation on a graph and its effect on the associated graph state is discussed. The effect of single-qubit Pauli measurements on graph states is discussed in sec. 3.4. Finally, sec. 3.5 concludes this chapter and gives further topics that can be studied.

The reader familiar with the theory behind graph states may feel free to skip this chapter.

3.1 | Graphs

A mathematical graph is, in its most basic form, a collection of points, that may or may not be connected to each other[1]. The points are generally referred to as *vertices* or *nodes*, and the latter name is used in this thesis. The connections between the nodes are called *edges*.

Definition 13. *A (simple) graph $G = (V, E)$ is a pair of two things:*

(1) *The vertex set V, a collection of nodes.*

(2) *The edge set E, a possibly empty collection of edges. An* edge *is a pair of two nodes.*

Edges cannot connect a node to itself, so it holds that $E \subset (V \times V) \setminus \{(i,i)\}_{i \in V}$. Any two elements $u, v \in V$ are said to be connected *if $(u, v) \in E$. The notation $V(G)$ refers to the vertex set V of the graph G.*

The collection $\mathcal{N}_u = \{v \in V | (v, u) \in E\}$ is the *neighbourhood* of u, i.e. the collection of nodes in V that are connected to u. A node i is *isolated* if it holds that $\mathcal{N}_i = \emptyset$.

A series of edges and nodes that link two nodes u and v is called a *path* between these two nodes. If there is a path between any two nodes in a graph, that graph is called *connected*. Unless explicitly stated otherwise, any graph in this thesis will be connected. A graph that is not connected is called *disconnected*, and consist of two or more smaller connected graphs. The

[1] In this thesis, only *simple, unweighted graphs* are considered, but they are referred to as just *graphs*.

reader that is familiar with graph theory may note that here only the *simple connected graphs* are considered.

Graphs have a clear graphical depiction, where circles represent nodes, and lines between them represent edges. Three examples of graphs are given in **Fig.** 3.1. They include four-node instances of two important types of graphs:

- The *line graph* L_V, the graph with edge set $\{(i, i+1)\}_{V\setminus\{n\}}$ that resembles a line[2].
- The *complete graph* K_V, with edge set $(V \times V) \setminus \{(i,i)\}_V$, i.e. the graph containing every possible edge.

In this thesis, any graph G will have a vertex set $V = [n] = \{1, 2, \dots n\}$, unless explicitly stated otherwise. n is referred to as the *size* of the graph, i.e. the number of vertices in G. Sometimes merely n is used as a shorthand for the vertex set; e.g. K_4 is the complete graph on the four nodes $V = \{1, 2, 3, 4\}$.

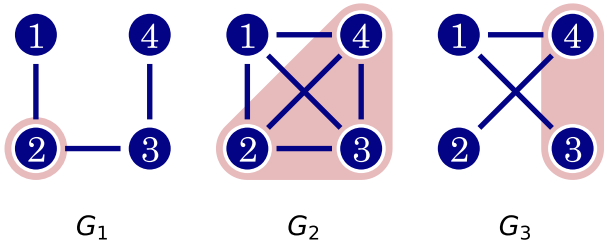


Figure 3.1: Three different graphs with four nodes each. The graph G_1 on the left is a *line graph* L_4, resembling a line from node 1 to node 4 through 2 and 3. The middle graph G_2 is the *complete graph* K_4, because it contains every possible edge. The graph G_3 on the right is a line graph as well, specifically the line $2-4-1-3$. The neighbourhood $\mathcal{N}_1$ of node 1 is highlighted in red for all three graphs. Finally, it holds that $G_1 = G_2 \oplus G_3$. Since $G_2 = K_4$ is the complete graph, G_1 and G_3 are each others' complimentary graph.

Given a graph $G = (V, E)$ and a node $i \in V$, the notation $G \setminus i$ indicates the graph that results from removing node i and all its incurrent edges. In other words, $G \setminus i = G' = (V', E')$, with

$$V' = V \setminus \{i\}, \tag{3.1}$$
$$E' = E \cap (V' \times V'). \tag{3.2}$$

[2]An ordering has been assumed to the vertex set V, with n being the last element. If this order is not apparent from context, or chosen differently, the order is made explicit: e.g. $G_r = L_{3142}$ from **Fig.** 3.1 has edge set $\{(3,1),(1,4),(4,2)\}$. This graph still represents a line, but on the path $3 \to 1 \to 4 \to 2$.

Given two random graphs $G = (V, E_G)$ and $H = (V, E_H)$, the graph $F = G \oplus H$ is the graph with vertex set V and edge set E_F, defined as the symmetric difference of the initial edge sets:

$$E_F = E_G \oplus E_H, \tag{3.3}$$

where $\oplus$ denotes the symmetric difference. In other words, E_F contains all edges that are either in E_G or in E_H, but not in both.

For any graph $G = (V, E)$, its *complementary graph* is the graph $G^\perp$ with vertices V and edge set $E^\perp = ((V \times V) \setminus \{(i, i)\}_V) \setminus E$, i.e. the graph with the same vertex set, that has exactly and only those edges that G does not have. An alternative definition is:

$$G^\perp = G \oplus K_V. \tag{3.4}$$

Finally, the *Adjacency matrix* $\Gamma \in \mathbb{F}_2^{n \times n}$ is a symmetric matrix that encodes the edge set:

$$\Gamma(i, j) = \begin{cases} 0 & (i, j) \notin E, \\ 1 & (i, j) \in E. \end{cases} \tag{3.5}$$

The columns of the adjacency matrix Γ encode the neighbourhood of the nodes. Let the i-th column of Γ be denoted as η_i; it is a vector of length n that has a 1 at entry j if node $j \in \mathcal{N}_i$, and 0 otherwise:

$$\eta_i(j) = \begin{cases} 0 & j \notin \mathcal{N}_i, \\ 1 & j \in \mathcal{N}_i. \end{cases} \tag{3.6}$$

3.1.1 | Local complementation

The *local complementation* is an important operation on a graph that is defined for each of its nodes, and transforms the graph into a new graph based on a graphical rule. A local complementation on node $i \in V$ is denoted τ_i, and the resulting graph is denoted $\tau_i(G)$. $\tau_i(G)$ results from the graph G where the subgraph on the neighbourhood $\mathcal{N}_i$ is replaced by its complementary graph. In other words, it results from G, where every possible edge between the elements of the neighbourhood $\mathcal{N}_i$ is inverted: the edge is removed or created if it was or wasn't there, respectively. Examples of local complementation are very instructive; for two examples see **Fig.** 3.2.

Using some slight abuse of notation, the local complementation can alternatively be defined in terms of a transformation of the adjacency matrix of a graph:

$$\Gamma_{\tau_i(G)} = \Gamma_G \oplus \Gamma_{K[\mathcal{N}_i]} = \Gamma_G \oplus \eta_i \eta_i^T \oplus \mathrm{diag}(\eta_i \eta_i^T), \tag{3.7}$$

where $K[\mathcal{N}_i]$ is the complete graph on the neighbourhood $\mathcal{N}_i$, and its adjacency matrix is assumed to have been 'extended' with the other nodes of the

graph G (i.e. the dimensions of the two adjacency matrices are compatible). The second equality follows from the fact that the outer product of η_i with itself resembles the complete graph on $\mathcal{N}_i$, except that the diagonal contains some 1's. From this definition it is evident that a local complementation is self-inverse: $\tau_i(\tau_i(G)) = G$.

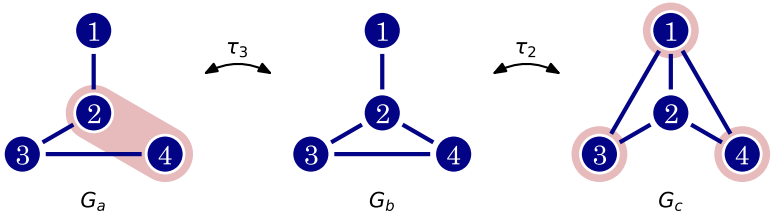


Figure 3.2: Two examples of local complementation. The graphs G_a and G_b are related by the local complementation τ_3 on node 3, so that $G_a = \tau_3(G_b)$. Similarly, $G_c = \tau_2(G_b)$, but there is no single node i such that $G_a = \tau_i(G_c)$. The neighbourhoods $\mathcal{N}_3$ and $\mathcal{N}_2$, that are inverted by the two local complementations, have been highlighted in G_a and G_c, respectively. A local complementation τ_i is self-inverse, so that $G_b = \tau_3(G_a)$ and $G_b = \tau_2(G_c)$.

Local complementations can be chained, so that graphs can be related by a series of local complementations. If two graphs G and G' are related by a local complementation τ_i, and G' is related to a third graph G'' by a local complementation τ_j, it follows that G and G'' are related by (at least) the combination of τ_i and τ_j. See for example G_a and G_c from **Fig.** 3.2, that are not related by a single local complementation, but are related as $G_c = \tau_2(\tau_3(G_a))$.

Because local complementations are self-inverse, they invoke an *equivalence relation*. When two graphs G and G' are related by a series of local complementations, they are called *locally equivalent*, denoted $G \sim G'$. This invites the definition of the *orbit*:

Definition 14. *For a given graph G, its orbit $\mathcal{O}(G)$ is the collection of all graphs H that are locally equivalent to G:*

$$\mathcal{O}(G) = \{H \text{ is a graph} | H \sim G\}. \tag{3.8}$$

Any element $H \in \mathcal{O}(G)$ is called a representative *of the orbit. The* size *of an orbit is the number of elements $|\mathcal{O}(G)|$ it contains.*

This definition partitions the set of all connected graphs of a fixed size: every (connected) graph belongs to exactly one orbit, and the collection of all orbits is exactly the set of all connected graphs. As an example, the complete orbit of L_4 can be found in **Fig.** 3.3. The graph L_{2143}, for instance, is in

the orbit, because it is related to L_4 by the successive local complementations τ_2, τ_1, τ_3 and τ_4, in that order.

Note that if two graphs are locally equivalent, it does not necessarily mean that the series of nodes to perform the local complementation on is unique. For instance, **Fig.** 3.3 shows that L_4 is additionally related to L_{2143} by the local complementations τ_3, τ_4, τ_2 and then τ_1.

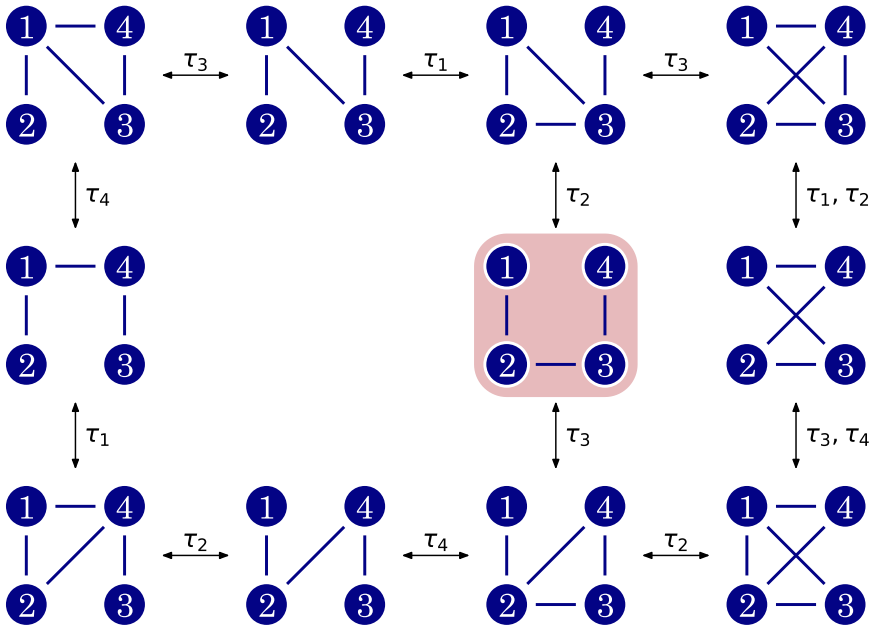


Figure 3.3: The entire orbit $\mathcal{O}(G_a)$ of the graph G_a from **Fig.** 3.2, shown in the middle and highlighted. $\mathcal{O}(G_a)$ is the collection of graphs that are *locally equivalent* to G_a: those graphs that can result from one or more successive local complementations performed on G_a. The graph G_b from **Fig.** 3.2 is part of the orbit, because it related to G_a by τ_3. Not all elements of the orbit are related to G_a by a single local complementation: e.g. G_c from **Fig.** 3.2 is related to G_a by two local complementations, namely τ_3 and τ_2. The different sequences from $G_a = L_4$ to the graph L_{2143} in the middle of the left column show that there may be multiple, distinct chains of local complementations that can link two graphs in an orbit.

3.2 | Graph states

An important subclass of the stabilizer states is formed by the *graph states*. They are a specific type of stabilizer state that can be defined in terms of a

graph.

Definition 15. *Let $G = (V, E)$ be a graph with $V = [n]$. The associated* graph state $|G\rangle$ *of G is the n-qubit state that results from initializing a qubit in the $|+\rangle$ state for every node in the graph, and applying a $C_Z^{(v,w)}$ operation between every pair of qubits v, w whose associated nodes share an edge:*

$$|G\rangle = \prod_{(v,w)\in E} C_Z^{(v,w)} |+\rangle^{\otimes V} . \tag{3.9}$$

All C_Z operations commute, so the order in which they are applied is irrelevant.

It is straightforward to show that any graph state is a stabilizer state. More specifically, the state $|+\rangle^{\otimes V}$ is a stabilizer state generated by $\{X_i\}_{i\in V}$; and only the Clifford operator C_Z (see (1.57)) is applied to it. It follows from the discussion in sec. 2.2 that $|G\rangle$ is a stabilizer state (see (2.12)).

Eq. (2.12) can also be used to determine the generators of a graph state $|G\rangle$. Denoting $U = \prod_{(v,w)\in E} C_Z^{(v,w)}$ and starting from the state $|+\rangle^{\otimes V}$, the generators are updated as:

$$X_i \to U X_i U^\dagger. \tag{3.10}$$

From (1.57) it follows that applying the operator $C_Z^{(i,w)}$ on X_i introduces an operator Z_w. The generator X_i is thus transformed by introducing a Z operator for every node that i shares an edge with, i.e. the neighbourhood $\mathcal{N}_i$ of i. The generators $\{g_i\}_{i=1}^n$ of a graph state $|G\rangle$ are therefore given by:

$$g_i = X_i \otimes \left(\bigotimes_{j\in\mathcal{N}_i} Z_j \right) = X_i Z_{\mathcal{N}_i}. \tag{3.11}$$

3.2.1 | Entanglement in graph states

From the underlying graph, it is straightforward to determine certain properties of graph states regarding its entanglement. Most notably, it is easy to see if a qubit in a graph state is separable: it is only separable from the rest of the state, if its associated node is isolated [3].

This extends naturally to multi-partite entanglement. More specifically, a graph state is multi-partite entangled (see (1.54)) if and only if its underlying graph is *connected* [3]. It is easy to determine if a graph is connected by e.g. a *breadth-first search* or by calculating its *algebraic connectivity* [97, 98]. This gives an efficient method to determine if a graph state is multi-partite entangled. From results like this, the study of multi-partite entanglement, as discussed in part II, is therefore greatly helped by the concept of graph states.

It should be noted that the number of edges of a graph does not represent the *amount* of entanglement in a graph state, measured by a suitable entanglement measure ([35], or see e.g. Def. 6); this will also follow from the results from sec. 3.3.

As mentioned before, in this thesis every graph is connected unless explicitly stated otherwise. Therefore, every associated graph state will be multi-partite entangled (unless explicitly stated otherwise).

3.2.2 | Examples of graph states

A straightforward but very important example of a graph state is given in **Fig.** 3.4. The graph $B = (\{1,2\},\{(1,2)\})$, i.e. the graph consisting of two, connected nodes, has the associated graph state:

$$|B\rangle = C_Z^{(1,2)} |+\rangle \otimes |+\rangle = \frac{1}{\sqrt{2}} (|0\rangle \otimes |+\rangle + |1\rangle \otimes |-\rangle) = (\mathbb{I} \otimes H) |B_{00}\rangle . \quad (3.12)$$

This shows that the Bell state $|B_{00}\rangle$ is, up to a local Clifford operation $\mathbb{I} \otimes H$, the graph state associated with two, connected, nodes. From (1.53) it follows that any other Bell state is, up to a local Clifford, the same graph state. Therefore, the connected two-node graph is often referred to as the *Bell-* or *EPR* pair.

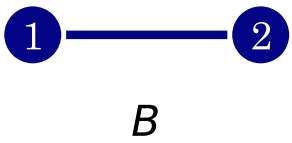


Figure 3.4: The graph B represents the graph state $|B\rangle = C_Z^{12} |+\rangle |+\rangle$; this is, up to a local Clifford operation, the Bell pair $|B_{00}\rangle$ (see (1.53)). Therefore, both B and $|B\rangle$ are often referred to as the *Bell pair* or *EPR pair*.

The linear cluster state

A somewhat more extensive example is given by the graph L_{123}, shown in **Fig.** 3.6. The associated graph state $|L_{123}\rangle$ is:

$$\begin{aligned} |L_{123}\rangle &= C_Z^{(1,2)} C_Z^{(2,3)} |+++\rangle \\ &= \frac{1}{\sqrt{2}} \left(C_Z^{(1,2)} C_Z^{(2,3)} |+0+\rangle + C_Z^{(1,2)} C_Z^{(2,3)} |+1+\rangle \right) \\ &= \frac{1}{\sqrt{2}} (|+0+\rangle + |-1-\rangle) . \end{aligned} \quad (3.13)$$

$|L_{123}\rangle$ is an example of a *linear cluster state*, which is the multi-partite entangled graph state associated with the line graph $L_{1\ldots n}$ for any number of nodes.

Definition 16. *The n-qubit linear cluster state $|L_n\rangle$ is the n-qubit graph state associated with the line graph $L_n = L_{1\ldots n}$. It can be written in the form*

$$|L_n\rangle = \prod_{i=1}^{n-1} CZ_{i,i+1} |+\rangle^{\otimes n} . \tag{3.14}$$

Furthermore, it has the canonical *generators*

$$g_i = \begin{cases} X_1 Z_2 & i = 1, \\ Z_{i-1} X_i Z_{i+1} & 2 \leqslant i \leqslant n-1, \\ Z_{n-1} X_n & i = n. \end{cases} \tag{3.15}$$

The GHZ state

Another important example is the GHZ state, named after *Greenberger*, *Horne* and *Zeilinger*, who introduced it in their seminal paper [99]. It is a multi-partite entangled state defined on n nodes, that is ubiquitous in many applications and facets of quantum- computation, communication and information theory. It can be seen as a generalisation of the Bell pair to more than two qubits.

Definition 17. *The (generalized) GHZ state $|\mathrm{GHZ}_n\rangle$ is the n-qubit stabilizer state defined as*

$$|\mathrm{GHZ}_n\rangle := \frac{1}{\sqrt{2}} \left(|0 \ldots 0\rangle + |1 \ldots 1\rangle \right) . \tag{3.16}$$

Furthermore, it has the canonical *generators*

$$g_i = \begin{cases} Z_i Z_{i+1} & 1 \leqslant i \leqslant n-1, \\ X_1 X_2 \ldots X_n & i = n. \end{cases} \tag{3.17}$$

The GHZ state is technically not a graph state, but it closely resembles the graph state associated with the *star graph*. The star graph is a graph with n nodes and the edge set $E = \{j, i\}_{i \in \{V \setminus j\}}$: the graph where a single node j, the *central node*, is connected to all other nodes, and no other edges exist. In the case that the central node is the first node, the graph state associated with the star graph is the state $\frac{1}{\sqrt{2}} \left(|0 + \cdots +\rangle + |1 - \cdots -\rangle \right)$. It follows that the GHZ state is related to the star graph state by a Hadamard operation on every node except the central node. Note that this operation is local Clifford.

The star graph is related to the complete graph K_n by a local complementation on the central node, and a subsequent local complementation on any other node results in a star graph centred around that node. The complete orbit of an n-node star graph thus consists of the complete graph and the n star graphs with the n different central nodes. The complete orbit of the 6-node star graphs is depicted in **Fig.** 3.5.

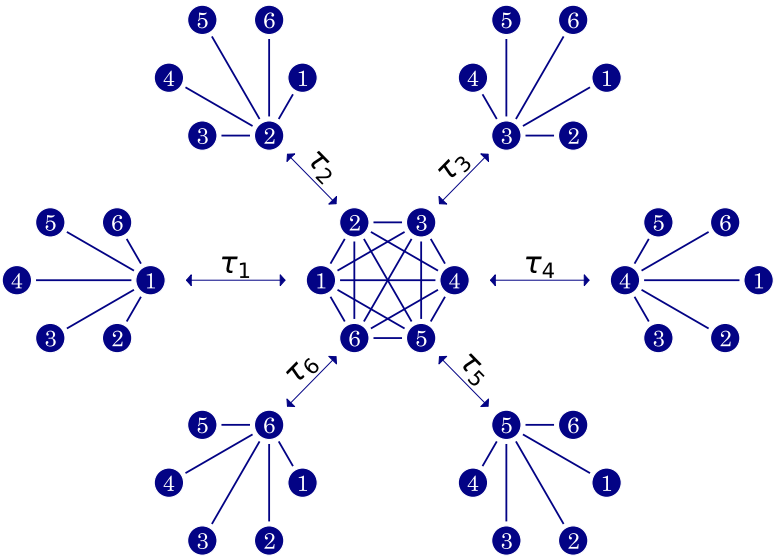


Figure 3.5: The *star graph* is the n-node graph in which a specific node, the *central node*, is connected to all other graphs, and all other graphs are only connected to the central node. It is related to to the complete graph K_n by a local complementation on the central node, so that its complete orbit is given by all n different star graphs, and K_n, which functions as a 'connection'. The star graph represents the graph state $\frac{1}{\sqrt{2}}(|0+\cdots+\rangle + |1-\cdots-\rangle)$ (where the central node is the first qubit). As such, it is closely related to the GHZ state $\frac{1}{\sqrt{2}}(|00\ldots0\rangle + |11\ldots1\rangle)$; the star- and complete graph are therefore synonymous with the GHZ state.

For reasons that will become apparent in the next section and in chapter 4, the n-qubit GHZ state is often taken synonymous with the star graph, and, because it is part of the same orbit, with the complete graph K_n.

3.3 Local complementations and local Clifford operations

When a local complementation τ_i is performed on a graph G, it is transformed to a graph $G' = \tau_i(G)$. The two associated graph states will be related by a unitary operation U_{τ_i}:

$$|G'\rangle = U_{\tau_i} |G\rangle . \tag{3.18}$$

There exists a strong correlation between local Clifford operations on graph states, and local complementations on the associated graphs. As shown in

this section, the unitary operator U_{τ_i} is in fact local Clifford, and can be determined by analysing the graph.

To do so and ease notation, it is useful to divide the neighbourhood $\mathcal{N}_i$ of the node i into four parts w.r.t. another node $j \in \mathcal{N}_i$:

- $\mathcal{N}_i^{\setminus j} = (\mathcal{N}_i \setminus \mathcal{N}_j) \setminus \{j\}$, the set of nodes that share an edge with node i but not with node j (except for node j itself).
- $\mathcal{N}_j^{\setminus i} = (\mathcal{N}_j \setminus \mathcal{N}_i) \setminus \{i\}$, the set of nodes that share an edge with node j but not with node i (except for node i itself).
- $\mathcal{N}_{i||j} = \mathcal{N}_i \cap \mathcal{N}_j$, the set of nodes that share an edge with both i and j.
- A set with only the node $\{j\}$.

It follows that $\mathcal{N}_i$ is the combination of all four sets:

$$\mathcal{N}_i = \{j\} \cup \mathcal{N}_i^{\setminus j} \cup \mathcal{N}_j^{\setminus i} \cup \mathcal{N}_{i||j}. \tag{3.19}$$

This notation can be used to perform a change of generators $\{g_j = X_j Z_{\mathcal{N}_j}\}$ of the graph state $|G\rangle$:

$$g_j \to g_j' = \begin{cases} g_j g_i = \left(X_j Z_{\mathcal{N}_j}\right)\left(X_i Z_{\mathcal{N}_i}\right) = Y_i Y_j Z_{\left(\mathcal{N}_i^{\setminus j}\right)} Z_{\left(\mathcal{N}_j^{\setminus i}\right)} \mathbb{I}_{\mathcal{N}_{i||j}} & j \in \mathcal{N}_i, \\ g_j = X_j Z_{\mathcal{N}_j} & j \notin \mathcal{N}_i, \end{cases} \tag{3.20}$$

where $\mathbb{I}_{\mathcal{N}_{i||j}}$ is written to emphasize that if $j \in \mathcal{N}_i$, g_j' does not have support on any node that shares an edge with both i and j.

When the local Clifford operation $U_{\tau_i} = \sqrt{X}_i^\dagger \sqrt{Z}_{\mathcal{N}_i} \in \mathcal{L}^{\mathcal{C}}$ is applied to the graph state, its generators are transformed as (see (2.12)):

$$g_j' \to U_{\tau_i} g_j' U_{\tau_i}^\dagger = \begin{cases} X_i Z_j Z_{\left(\mathcal{N}_i^{\setminus j}\right)} Z_{\left(\mathcal{N}_j^{\setminus i}\right)} & j \in \mathcal{N}_i, \\ X_j Z_{\mathcal{N}_j} & j \notin \mathcal{N}_i. \end{cases} \tag{3.21}$$

These are exactly the generators of the graph state $|\tau_i(G)\rangle$, i.e. the state associated with the graph $\tau_i(G)$. It can be concluded that for any pair of graphs G and $G' = \tau_i(G)$, their associated graph states $|G\rangle$ and $|G'\rangle$ are related by a local Clifford operation $U_{\tau_i} = \sqrt{X}_i^\dagger \sqrt{Z}_{\mathcal{N}_i}$:

$$|G'\rangle = |\tau_i(G)\rangle = U_{\tau_i} |G\rangle . \tag{3.22}$$

Fig. 3.6 contains three examples. The graph states $|L_{213}\rangle$, $|L_{123}\rangle$ and $|L_{132}\rangle$ are all related to $|K_3\rangle$ by a local Clifford operation U_{τ_1}, U_{τ_2} and U_{τ_3}, respectively. This can be understood by the fact that their associated graphs are all related to K_3 by a local complementation on those same nodes (note

	c.o.g.		U_{τ_2}	
XZZ	$\rightarrow$	YYI	$\rightarrow$	XZI
ZXZ	$\rightarrow$	ZXZ	$\rightarrow$	ZXZ
ZZX	$\rightarrow$	IYY	$\rightarrow$	IZX

Table 3.1: The generators of the graph state $|K_3\rangle$ from **Fig.** 3.6 are first changed (labelled **c.o.g.**) towards another set, after which they are transformed under the local Clifford unitary $U_{\tau_2} = \sqrt{X}_2^\dagger \sqrt{Z}_1 \sqrt{Z}_3$. This results in the generators of the graph state $|L_{123}\rangle$.

that these graphs all belong to the 3-node GHZ orbit). **Tab.** 3.1 shows how the generators of $|K_3\rangle$ relate to those of $|L_{123}\rangle$ in more detail.

The relation in (3.22) gives a clear graphical rule for the effect of the local Clifford U_{τ_i} on any graph state G. A (stronger) reverse statement is true as well, which will be presented and discussed in chapter 4.

Finally, note that $U_{\tau_i}^2 = g_i$, reflecting the fact that a local complementation is self-inverse. Moreover, the unitary operator implemented by a local complementation is not unique: a rotation in the different direction for both the Z and X axis works as well. As such, there are two equivalent options for the local Clifford U_{τ_i} that represent the local complementation:

$$U_{\tau_i} = \begin{cases} \sqrt{X}_i^\dagger \sqrt{Z}_{(\mathcal{N}_i)}, \\ \sqrt{X}_i \sqrt{Z}_{(\mathcal{N}_i)}^\dagger. \end{cases} \tag{3.23}$$

The difference between these two operators is exactly the generator g_i, which acts as the identity $\mathbb{I}$ on the graph state because it is a stabilizer element.

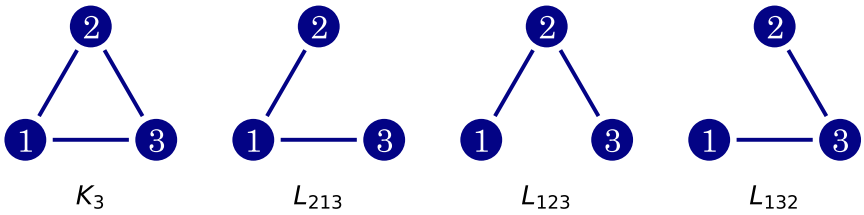


Figure 3.6: The complete graph K_3 is associated with the graph state $|K_3\rangle = C_Z^{1,2} C_Z^{1,3} C_Z^{2,3} |+++\rangle$. Following (3.13), the graph L_{123} represents the state $|L_{123}\rangle = \frac{1}{\sqrt{2}} (|+0+\rangle + |-1-\rangle)$. Since K_3 and L_{123} are related by a local complementation τ_2, it holds that $|K_3\rangle = U_{\tau_2} |L_{123}\rangle = \sqrt{X}_1^\dagger \sqrt{Z}_2 \sqrt{Z}_3 |L_{123}\rangle$. Similarly, it holds that $|K_3\rangle = U_{\tau_1} |L_{213}\rangle$ and $|K_3\rangle = U_{\tau_3} |L_{132}\rangle$.

3.4 | Single-qubit Pauli measurements on graph states

The action of single-qubit Pauli basis measurements on graph states can be understood in terms of the underlying graph. Especially the result of a Z-basis measurement is straightforward, but the Y- and X-basis measurements can be understood in terms of the underlying graph as well. The Z-, Y- and X-basis measurements are discussed in secs. 3.4.1 to 3.4.3, respectively.

3.4.1 | Measurement of a single node in the Z basis

The measurement outcome of a measurement in the Z basis on node i is straightforward: there is only one generator that does not commute with Z_i, namely the generator $g_i = X_i Z_{\mathcal{N}_i}$. Therefore, by (2.19), the measurement outcome $m = \pm 1$ is uniformly random.

Because there is only one generator that anti-commutes with the measurement operator Z_i, it is straightforward to determine the post-measurement state using the results presented in sec. 2.3.1. It follows that the resulting post-measurement state is stabilized by the generators $\{g_j\}_{j \neq i}$, and that the generator g_i is replaced by the observable $(m)Z_i$, that now carries a phase.

In a networked setting, any qubit that has been measured is not useful afterwards, so only the post-measurement state of the rest of the nodes is important. The measured qubit can be removed from the generators using the method from e.g. **Tab.** 2.2. Following the same analysis, the measurement outcome is introduced as a phase for every generator that is associated with a node in the neighbourhood of the measured node.

The generators of this $(n-1)$-qubit post-measurement state are:

$$g_j = \begin{cases} X_j Z_{\mathcal{N}_j} & j \notin \mathcal{N}_i, \\ (m) X_j Z_{\mathcal{N}_j} & j \in \mathcal{N}_i. \end{cases} \tag{3.24}$$

When the measurement outcome is $m = +1$, this is exactly the graph state $|G \setminus i\rangle$ of the graph $G \setminus i$, i.e. the graph with node i deleted (see (3.1)).

In the case that the measurement outcome is $m = -1$, the second set of generators carry a non-trivial phase, so that the post-measurement state is not a graph state. However, the local Clifford operation $Z_{\mathcal{N}_i}$ removes this non-trivial phase, so that the post-measurement state then becomes the graph state $|G \setminus i\rangle$ as well.

In conclusion, there are clear and straightforward graphical rules for the effect of a Z-basis measurement on a graph state. The post-measurement state is the graph state $|G \setminus i\rangle$, i.e. the graph with the measured node removed, up to a local Clifford correction $Z_{\mathcal{N}_i}$ when the measurement outcome m equals -1. Two examples are given in **Fig.** 3.7.

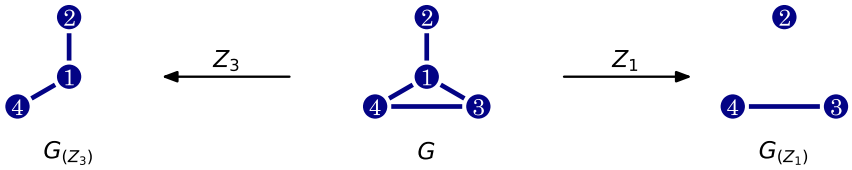


Figure 3.7: The graph in the middle is the graph $G = G_b$ from **Fig.** 3.2. Measuring a node of a graph state in the Z-basis results in a graph state with that node removed. Thus, measuring node 3 of G results in the three-qubit graph state $|G_{(Z_3)}\rangle$. Similarly, measuring node 1 of G results in the three-qubit graph state $|G_{(Z_1)}\rangle$. The choice of node for Z-basis measurements can greatly influence the post-measurement state: the graph state $|G_{(Z_3)}\rangle$ is the same as $|L_{412}\rangle$, but for $|G_{(Z_1)}\rangle$, the second node 2 is completely disentangled from nodes 4 and 3, that form a Bell pair. The actual post-measurement states may not be exactly the depicted graph states, but can differ by a (measurement-outcome-dependent) local Clifford rotation.

3.4.2 | Measurement of a single node in the Y basis

A measurement in the Y basis on i follows from the analysis of the Z-basis measurement; in particular, graphical rules can be obtained as well. There is always at least one generator that anti-commutes with Y_i (namely $g_i = X_i Z_{\mathcal{N}_i}$), so by (2.19) the measurement outcome $m = \pm 1$ is uniformly random.

The post-measurement state for either outcome can additionally be determined. Using the relations $|+i\rangle = \sqrt{X}^\dagger |0\rangle$ and $|-i\rangle = \sqrt{X}^\dagger |1\rangle$, a Y-basis measurement can be seen as a Z-basis measurement preceded by the Clifford operator $\sqrt{X}$. This insight can be used to determine the post-measurement state for the outcome $m = +1$:

$$\begin{aligned} |+i\rangle\langle+i|_i |G\rangle &= \sqrt{X}_i^\dagger |0\rangle\langle 0|_i \sqrt{X}_i |G\rangle \\ &= \sqrt{X}_i^\dagger |0\rangle\langle 0|_i \sqrt{X}_i \underbrace{\sqrt{Z}_{\mathcal{N}_i}^\dagger \sqrt{Z}_{\mathcal{N}_i}}_{\mathbb{I}} |G\rangle \\ &= \sqrt{Z}_{\mathcal{N}_i} \sqrt{X}_i^\dagger |0\rangle\langle 0|_i \underbrace{\sqrt{X}_i \sqrt{Z}_{\mathcal{N}_i}^\dagger}_{U_{\tau_i}} |G\rangle \\ &= \sqrt{Z}_{\mathcal{N}_i} \sqrt{X}_i^\dagger |0\rangle\langle 0|_i |\tau_i(G)\rangle . \end{aligned} \tag{3.25}$$

The analysis for the $m = -1$ outcome follows similarly.

Hence, up to a local Clifford rotation $\sqrt{Z}_{\mathcal{N}_i}$, a measurement of a node i on a graph state $|G\rangle$ in the Y-basis acts the same as a Z-basis measurement of the same node on the graph state $|\tau_i(G)\rangle$.

A Z-basis measurement involves deleting the node, so that the post-measurement state is $|\tau_i(G) \setminus i\rangle$, up to a local Clifford operation. For two examples, see **Fig.** 3.8.

This local Clifford operation consists of $Z_{\mathcal{N}_i}$ when the measurement outcome is -1, followed by the correction $\sqrt{Z}_{\mathcal{N}_i}$ regardless of the measurement outcome. Note that these two operators commute, so they can be applied in either order.

In conclusion, a measurement of Y_i on an n-qubit graph state $|G\rangle$ results in the $(n-1)$-qubit graph state $|\tau_i(G) \setminus i\rangle$. However, note that the post measurement state for neither the $+1$ nor -1 outcome is this exact graph state, but merely a stabilizer state related by a local Clifford operation to the one given in the analysis.

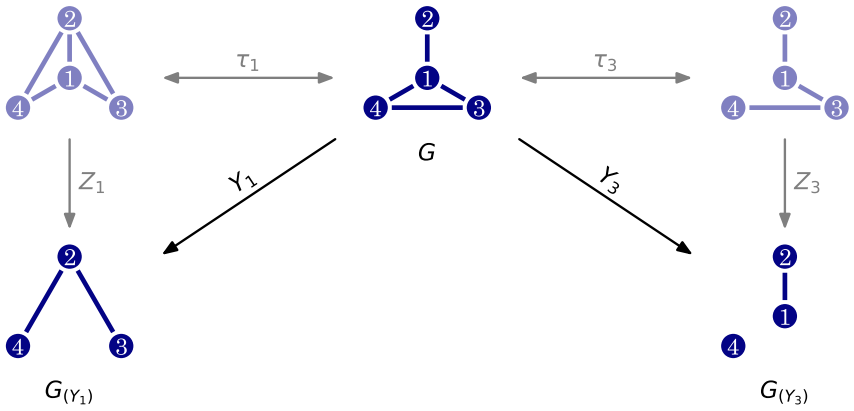


Figure 3.8: Similarly to Z-basis measurements, the effect of Y-basis measurements on graph states can be analysed using graphical methods. Measuring a node of a graph state in the Y basis can be interpreted as performing a Z-basis measurement on the same node, preceded by a local complementation on that node. Thus, measuring node 1 of G in the Y basis results in the three-qubit graph state $|G_{(Y_1)}\rangle$, which can be obtained by first applying a local complementation τ_1 and subsequently removing node 1: $G_{(Y_1)} = \tau_1(G) \setminus 1$. Similarly, measuring node 3 of G in the Y basis results in the three-qubit graph state $|G_{(Y_3)}\rangle = |\tau_3(G) \setminus 3\rangle$. As with Z-basis measurements, the choice of node for Y-basis measurements can greatly influence the post-measurement state, resulting in very different states that may be entangled or not. Note that all the post-measurement states are only the depicted graph states up to a (measurement-outcome-dependent) local Clifford rotation.

3.4.3 | Measurement of a single node in the X basis

Similarly to a measurement in the Y basis, graphical rules for a measurement in the X basis on a node i follows from a Z-basis measurement. There

is always at least one generator that anti-commutes with X_i (namely any generator associated with a node in $\mathcal{N}_i$), so that by (2.19) the measurement outcome $m = \pm 1$ is uniformly random. Again, the post-measurement state for either outcome can be determined. Similar to the Y basis, the identities $|+\rangle \hat{=} \sqrt{Z}^\dagger \sqrt{X}^\dagger |0\rangle$ and $|-\rangle \hat{=} \sqrt{Z}^\dagger \sqrt{X}^\dagger |1\rangle$ can be used[3]. This allows the X-basis measurement to be represented by a Z-basis measurement, preceded by a rotation by the local Clifford operator $\sqrt{X}\sqrt{Z}$.

The rotation operator $\sqrt{Z}^\dagger \sqrt{X}^\dagger$ is somewhat more involved than for the Y-basis case, and cannot be realised by a single local complementation. However, a local complementation on any node in the neighbourhood $\mathcal{N}_i$ of i can induce a $\sqrt{Z}$ rotation on node i, and a local complementation τ_i on node i itself can induce a $\sqrt{X}$ rotation on node i; combining these two can realise the necessary rotation.

To this effect, let $k \in \mathcal{N}_i$ be a random neighbour of i. A local complementation on k, followed by a local complementation on i gives the state:

$$\begin{aligned}
|\tau_i(\tau_k(G))\rangle &= U_{\tau_i} U_{\tau_k} |G\rangle \\
&= U_{\tau_i} \sqrt{X}_k^\dagger \sqrt{Z}_{\mathcal{N}_k} |G\rangle \\
&= U_{\tau_i} \sqrt{X}_k^\dagger \sqrt{Z}_{(\mathcal{N}_k \setminus \{i\})} \sqrt{Z}_i |G\rangle \\
&= \sqrt{X}_i \sqrt{Z}_{\mathcal{N}_i}^\dagger \sqrt{X}_k^\dagger \sqrt{Z}_{(\mathcal{N}_k \setminus \{i\})} \sqrt{Z}_i |G\rangle \\
&= \sqrt{Z}_{\mathcal{N}_i}^\dagger \sqrt{X}_k^\dagger \sqrt{Z}_{(\mathcal{N}_k \setminus \{i\})} \sqrt{X}_i \sqrt{Z}_i |G\rangle \\
&= A \sqrt{X}_i \sqrt{Z}_i |G\rangle ,
\end{aligned} \tag{3.26}$$

where $A = \sqrt{Z}_{\mathcal{N}_i}^\dagger \sqrt{X}_k^\dagger \sqrt{Z}_{(\mathcal{N}_k \setminus \{i\})}$ is a local Clifford operation.

For the measurement outcome $m = +1$, a measurement in the X basis then results in the post-measurement state:

$$\begin{aligned}
|+\rangle\langle+|_i |G\rangle &= \sqrt{Z}_i^\dagger \sqrt{X}_i^\dagger |0\rangle\langle 0|_i \underbrace{\sqrt{X}_i \sqrt{Z}_i |G\rangle}_{A^\dagger |\tau_i(\tau_k(G))\rangle} \\
&= A^\dagger \sqrt{Z}_i^\dagger \sqrt{X}_i^\dagger |0\rangle\langle 0|_i |\tau_i(\tau_k(G))\rangle .
\end{aligned} \tag{3.27}$$

The post-measurement state of the $m = -1$ outcome follows similarly.

To ease notation, let $G' = \tau_i(\tau_k(G)) \setminus i$ be the graph obtained after the two local complementations and the node deletion. When the measured node is removed, the post-measurement state is $A^\dagger |G'\rangle$ or $A^\dagger Z_{\mathcal{N}_i} |G'\rangle$, for the $+1$ and -1 outcome, respectively.

Although this gives a closed form for the post-measurement state of either measurement outcome, some extra insights can be instructive. Node i was

[3] Note that these identities are only true up to an irrelevant phase, which will cancel out for the measurement operators $|+\rangle\langle+|$ and $|-\rangle\langle-|$ (see (3.27)).

removed, so it can be dropped from $A^\dagger$, resulting in $A^\dagger = \sqrt{Z}^\dagger_{\mathcal{N}_k}\sqrt{X}_k\sqrt{Z}_{\mathcal{N}_i} = U_{\tau_k}\sqrt{Z}_{\mathcal{N}_i}$ (i.e. node i is now understood to be removed from $\mathcal{N}_k$).

Moreover, the identity $\sqrt{X}\sqrt{Z}\sqrt{X}^\dagger = i\sqrt{Y}^\dagger$ implies that:

$$U_{\tau_k}\sqrt{Z}_{\mathcal{N}_i} = i\sqrt{Z}_{(\mathcal{N}_i\setminus\{k\})}\sqrt{Y}^\dagger_k U_{\tau_k}. \tag{3.28}$$

This can be used to interpret part of $A^\dagger$ as another local complementation. Specifically for the $+1$ measurement outcome, this results in (up to a irrelevant phase):

$$\begin{aligned} A^\dagger |G'\rangle &= U_{\tau_k}\sqrt{Z}_{\mathcal{N}_i}|G'\rangle \\ &= \sqrt{Z}_{(\mathcal{N}_i\setminus\{k\})}\sqrt{Y}^\dagger_k U_{\tau_k}|G'\rangle \\ &= \sqrt{Z}_{(\mathcal{N}_i\setminus\{k\})}\sqrt{Y}^\dagger_k |\tau_k(G')\rangle\,. \end{aligned} \tag{3.29}$$

The -1 outcome can be addressed similarly. The identity $\sqrt{X}\sqrt{Z}^\dagger\sqrt{X}^\dagger = i\sqrt{Y}$ implies:

$$U_{\tau_k}\sqrt{Z}^\dagger_{\mathcal{N}_i} = i\sqrt{Z}^\dagger_{(\mathcal{N}_i\setminus\{k\})}\sqrt{Y}_k U_{\tau_k}, \tag{3.30}$$

which can be used to show that for the $m = -1$ outcome, the post-measurement state is $\sqrt{Z}^\dagger_{(\mathcal{N}_i\setminus\{k\})}\sqrt{Y}_k|\tau_k(G')\rangle$ (up to an irrelevant phase).

Eqs. (3.29) and (3.30) show that performing another local complementation on the (randomly chosen but fixed) node k after the removal of the measured node i results in a graph state that is in essence 'closer' to the true post-measurement state: the local Clifford correction consists of fewer non-trivial operations. For this reason, the last local complementation is often included in the analysis of an X-basis measurement.

In conclusion, an X-basis measurement on node i of a graph G results in a post-measurement state that is, up to a (measurement-outcome dependent) local Clifford rotation the following graph state:

$$|\tau_k(\tau_i(\tau_k(G))\setminus i)\rangle\,, \tag{3.31}$$

where k is a random node from the neighbourhood $\mathcal{N}_i$ of node i. The measurement-outcome dependent local Clifford operation can be retrieved from (3.29) and (3.30). Finally, note that an X-basis measurement can also be understood as a local complementation on the node k, followed by a Y-basis measurement. See **Fig.** 3.9 for an example of an X-basis measurement, but note that there is some ambiguity in choosing the node k.

Freedom in choosing k

The choice of the node k from the neighbourhood $\mathcal{N}_i$ is arbitrary, but different choices may result in different graphs. As an example, the graph G from **Fig.** 3.8 has $\mathcal{N}_4 = \{1,3\}$. **Fig.** 3.9 shows the post measurement states resulting from an X-basis measurement on node 4 for this graph for

both $k = 1$ and $k = 3$ as the random choice of node in the neighbourhood. The resulting graph states are not the same. However, the bottom row of **Fig.** 3.9 shows that the graphs are related by two local complementations.

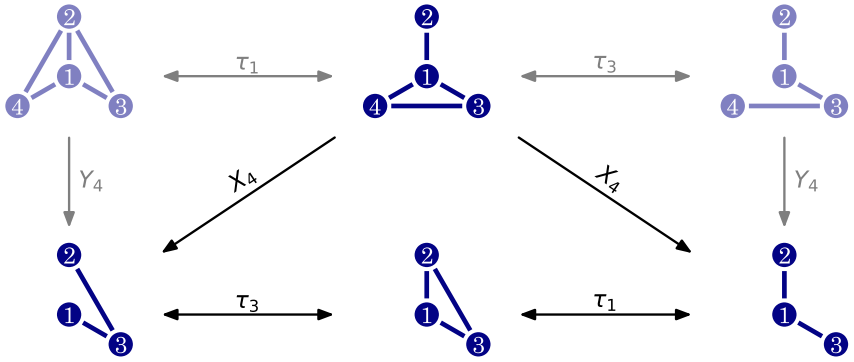


Figure 3.9: The effect of an X-basis measurement on a graph state. An X-basis measurement of a node i on a graph state $|G\rangle$ results in the graph state $|\tau_k(\tau_i(\tau_k(G)) \setminus i)\rangle$, where k is a random neighbour of i. The choice of neighbour k is not trivial, as the example shows. Choosing node 1 or node 3 as the random neighbour for an X-basis measurement of node 4 results in two different post-measurement graphs. Nevertheless, these two graphs are locally equivalent, a fact which will be true for any choice of X-basis measurement on any graph state.

This is exemplary of a more general fact. Eq. (3.27) and the analysis directly after it determines that the post-measurement state is $A_k^\dagger |\tau_i(\tau_k(G)) \setminus i\rangle$, for any choice of neighbour k (the correction operator $A = A_k$ now carries a subscript k to emphasize that it is dependent on the choice of k). Equating the post measurement states for two different k and k' results in:

$$\begin{aligned} A_k^\dagger |\tau_i(\tau_k(G)) \setminus i\rangle &= A_{k'}^\dagger |\tau_i(\tau_{k'}(G)) \setminus i\rangle \\ &\rightarrow |\tau_i(\tau_k(G)) \setminus i\rangle = A_k A_{k'}^\dagger |\tau_i(\tau_{k'}(G)) \setminus i\rangle \end{aligned} \tag{3.32}$$

Both $A_{k'}^\dagger$ and A_k are local Clifford, so their product is as well, which means that the graph states $|\tau_i(\tau_k(G)) \setminus i\rangle$ and $|\tau_i(\tau_{k'}(G)) \setminus i\rangle$ are related by a local Clifford operation.

Fig. 3.9 shows that for the two particular post-measurement states that it contains, their associated graphs are locally equivalent (i.e. they are in each others orbit). However, this does not follow immediately for the general case. Is it guaranteed that the two graphs $\tau_i(\tau_k(G)) \setminus i$ and $\tau_i(\tau_{k'}(G)) \setminus i$ are locally equivalent because their associated graph states $|\tau_i(\tau_k(G)) \setminus i\rangle$ and $|\tau_i(\tau_{k'}(G)) \setminus i\rangle$ are related by a local Clifford operation? This is an important

question that is addressed in chapter 4, and is the reverse statement that was hinted at earlier.

3.5 | Conclusion and further reading

As was explained in sec. 2.4, the entanglement properties of stabilizer states can be analysed by their reduced states. Some properties of the marginals of graph states can be inspected from their associated graph alone, so that e.g. the entanglement entropy (see Def. 6) of a graph state can be computed by inspecting the graph. These marginals and their properties are addressed in more detail in chapter 6, where they are used to study multi-partite entanglement.

Like with the stabilizer formalism (see sec. 2.5), there exist extensions of the theory of graph states so that a larger set of states can be represented. The most well known such extension defines states in terms of *hyper graphs*, where an edge is not necessarily a pair (v, w), but can be a *hyper edge*, i.e. any subset of V. The associated *hyper graph state* [100] is then defined in a similar manner to the standard graph state, where the C_Z gate in its definition is extended to the generalized multi-controlled version $C_Z^{\otimes n}$. In this n-qubit gate, a Z gate is applied to the last qubit if and only if the state of all other qubits is $|1\rangle$. However, this generalized C_Z gate is generally not Clifford, so a hyper-graph state usually fails to be a stabilizer state.

Another well-known extension is formed by the *weighted graph states* [3, 101, 102], represented by graphs whose edges carry *weights*, i.e. any real number $\phi \in [0, 2\pi)$. For such weighted graph states, the C_Z gate is replaced by a controlled phase gate $\mathrm{diag}(1, 1, 1, e^{i\phi})$. Such a gate is Clifford only for $\phi \in \{0, \pi\}$, so it follows that most weighted graph states are not stabilizer states.

Graph states play an important role in the theory of quantum communication and networks. In these networked settings, the only operations that are freely available are the *local operations*, e.g. a local unitary, or a single-qubit measurement. Multi-partite entanglement, the topic of part II, is therefore characterized by local operations. Chapter 4 makes various of their concepts and notions more precise, and discusses the *equivalence* of stabilizer states under these local operations.

PART II

Multi-partite Entanglement in Quantum Networks

4

Local operations on stabilizer states

All separable states are alike;
each entanglement class is
entangled in its own way.

Leo Tolstoy, *Anna Karenina*
(paraphrased)

Graph states are ubiquitous in quantum networking settings, and play an important role in many communication protocols. In these networked settings, it is usually assumed that every node has one qubit, and that all n qubits are together in some (entangled) state. Operations that involve multiple qubits at the same time (e.g. a C_Z or C_X gate) are generally hard to implement in a network, because they involve the communication of quantum signals (e.g. the qubits have to be 'brought together' to implement a C_Z gate).

At the same time, operations on single qubits, like local unitary operators and single-qubit measurements, are much easier to implement. Although they cannot create entanglement, these *local operations* can have a non-trivial effect on the total quantum state of the network, so that two ostensibly different stabilizer states can be *locally equivalent*.

Part II studies the local equivalence of pure, multi-partite entangled states, i.e. the equivalence of a multi-qubit state $|\psi_2\rangle$ and another multi-qubit state

$|\psi_1\rangle$ under single-qubit operations. In this and the following chapters, $|\psi_1\rangle$ will be referred to as the *resource state*, and $|\psi_2\rangle$ will be referred to as the *target state*, especially when measurements are considered. As is customary in quantum communication, only stabilizer and graph states are considered.

Part II consists of chapters 4 to 6; chapter 4, this chapter, introduces the relevant concepts and results from literature. Chapters 5 and 6 present the contents of Pubs. **[F]** and **[G]**, respectively, and will be introduced in the conclusion of this chapter.

In principle, the term 'locally equivalent' indicates the case where the states $|\psi_1\rangle$ and $|\psi_2\rangle$ have an equal number of qubits, so that no measurements are involved. In such a setting the operations can usually be inverted, so that $|\psi_1\rangle$ can be obtained from $|\psi_2\rangle$, indicating a proper *equivalence relation* between the states. This raises an important question: given two different states, can it be determined if they are locally equivalent or not, and if so, by what operations? Moreover, given a specific state, what is the set of other states that are 'reachable' by local operations?

The more general case, involving measurements, is not necessarily invertible. Here, the number of qubits of $|\psi_2\rangle$ is potentially lower than the number of qubits of $|\psi_1\rangle$, so that reconstructing the state $|\psi_1\rangle$ from the state $|\psi_2\rangle$ is impossible without providing new qubits and, potentially, multi-qubit gates. Consider e.g. **Tab.** 2.2, where one qubit of a Bell state is measured in the X basis. The post-measurement state is separable, so it can not be locally equivalent to the original Bell state. It follows that by including measurements, an equivalence relation is not obtained. Although the setting is nevertheless important from an operational point of view, it is less well understood, and many of the relevant questions are harder to answer.

Chapter 4 introduces various important results that intricately come together to determine many aspects of local equivalence of stabilizer states, both in the setting without, and with measurements. More specifically, in sec. 4.1 the setting of local equivalence is made more precise, and the *Local Operations and Classical Communication* (*LOCC*) paradigm and other related paradigms are introduced. Additionally, the section introduces the first important result, which shows that for the local equivalence of stabilizer states of equal size, one can focus solely on local unitary operations, instead of a much more broader class of operations.

A second important result is presented in sec. 4.2, which shows that every stabilizer state is locally equivalent to at least one graph state. This means that in the study of local equivalence of stabilizer states, one needs to consider graph states only.

Section 4.3 presents various sets of graph states that are grouped under different notions of equivalence, which are useful to discuss local equivalence more precisely.

The equivalence of graph states of equal size is discussed in sec. 4.4. More specifically, a third important result is presented in sec. 4.4.1, which shows an

intricate interplay between the equivalence of graph states under local Clifford operations, and local (complementation) equivalence of the associated graphs. The same section additionally presents the fourth important result, which is an efficient method to determine if two graph states are equivalent under local Clifford operations. The difference between the equivalence of graph states under local Clifford and local unitary operations is then discussed in sec. 4.4.2.

The setting where measurements are included is discussed in sec. 4.5. Finally, the chapter is concluded in sec. 4.6, where additionally the other chapters of part II are introduced.

The reader familiar with the theory of local equivalence of graph states may feel free to skip this chapter, although there doesn't exist standard notation for some of the concepts introduced in sec. 4.3; some details presented in that section regarding the number of LU-orbits per entanglement class are not found in literature either. Additionally, the results from **Tab.** 4.1 are calculated by me, and (technically) use the results of Pub. **[G]** where it is shown that the number of *LC-classes* and *entanglement classes* (see sec. 4.3) is equal for nine qubits.

4.1 | Local operations and the LOCC paradigm

When two quantum systems A and B are entangled, the system of A cannot be specified without including B as well, and vice versa. As sec. 1.5 explained, this implies that measurements on one system can collapse the state of the system. Consider the Bell pair $|B_{00}\rangle$, and an X-basis measurement on the first qubit (see **Tab.** 2.2). If the outcome m_1 of the measurement is 0 or 1, the second qubit has collapsed to $|+\rangle$ or $|-\rangle$, respectively.

However, if B does not learn the measurement outcome, the state ρ_2 for the second qubit is a statistical mixture between these two collapsed states:

$$\rho_2 = \Pr(m_1 = 0)\,|+\rangle\langle+| + \Pr(m_1 = 1)\,|-\rangle\langle-| = \frac{\mathbb{I}}{2}, \tag{4.1}$$

where the last equality follows because the two outcomes are equally likely. System B has statistical ambiguity regarding the state of the qubit, which can only be removed by learning the measurement outcome. Because the outcome itself is classical, it can be communicated by *classical communication*. This shows that it is important to include classical communication when considering local operations.

The inclusion of classical communication in the set of allowed operations is made rigorous by the paradigm called *local operations and classical communication*, often abbreviated as *LOCC* [35, 36]. A complete introduction is beyond the scope of this thesis, but an LOCC operation is essentially a general quantum channel (see (1.30)), where the map Λ can be written in a certain *separable* form. For the purpose of this thesis, it is not important to specify LOCC operations further.

Technically, the LOCC paradigm only permits deterministic transformations. The generalisation where probabilistic transformations are allowed is known as *stochastic* LOCC or *SLOCC*. The study of bi-partite entanglement, and if one state can be transformed to another, is largely covered by the SLOCC paradigm [35, 36]. One of the main results in entanglement theory was proven by Nielsen in [103], which states that a bi-partite (entangled) state $|\psi_1\rangle$ can be transformed to another bi-partite state $|\psi_2\rangle$ by SLOCC operations if and only if the Schmidt coefficients (see Def. 5) of $|\psi_2\rangle$ are *majorized* [36] by those of $|\psi_1\rangle$.

Other paradigms of local operations have been studied as well. In networking scenarios, it might not always be possible or practical to perform classical communication, especially with current levels of quantum hardware[1]. For that reason, it has been considered to reduce the set of allowed operations to so-called *local operations and shared randomness* or *LOSR* [104]. In this paradigm the nodes in the network are not able to classically communicate, but they do have access to shared randomness, which they can use to take decisions regarding the transformation. Although not widely studied or well understood, recently some no-go results have been shown regarding LOSR. In particular, it is not possible to prepare graph states using only Bell pairs in an LOSR setting [53, 54], which was shown using *inflation techniques* [105].

Reducing the set of allowed operations even further, the paradigm of *LO* (for *local operations*) permits only local operations, without any coordination made possible by communication or shared randomness. In the study of (multi-partite entanglement) equivalence, this is often restricted further to include only *local unitary* operations, which results in the notion of *LU*-equivalence.

Definition 18. *Let $|\psi_1\rangle$ and $|\psi_2\rangle$ be two n-qubit quantum states. $|\psi_1\rangle$ and $|\psi_2\rangle$ are* LU-equivalent *if there exists a local unitary operation $U \in \mathcal{L}_n^{\mathcal{U}}$ such that:*

$$|\psi_2\rangle = U\,|\psi_1\rangle\,. \tag{4.2}$$

Because $U^\dagger \in \mathcal{L}_n^{\mathcal{U}}$ if and only if $U \in \mathcal{L}_n^{\mathcal{U}}$, and $UV \in \mathcal{L}_n^{\mathcal{U}}$ for any $U, V \in \mathcal{L}_n^{\mathcal{U}}$, an equivalence relation is implied.

The first important result that was mentioned in the introduction is regarding the equivalence of graph states under SLOCC or under local unitary operators. It holds, perhaps surprisingly, that multi-partite entangled graph states are SLOCC equivalent if and only if they are LU-equivalent. This was shown in [106], using results from [107], and follows from the fact that all single-qubit marginal states of (connected) graph states are maximally mixed. It follows that it suffices to consider just local unitary operations when investigating the equivalence of graph states.

[1] The quantum states could e.g. have decohered before the classical communication arrives.

Finally, instead of considering all local unitary operations, one can in fact restrict to only local Clifford operations.

Definition 19. *Let $|\psi_1\rangle$ and $|\psi_2\rangle$ be two n-qubit quantum states. $|\psi_1\rangle$ and $|\psi_2\rangle$ are* LC-equivalent *if there exist a local Clifford operation $C \in \mathcal{L}_n^{\mathcal{C}}$ such that:*

$$|\psi_2\rangle = C\,|\psi_1\rangle\,. \tag{4.3}$$

Because $C^\dagger \in \mathcal{L}_n^{\mathcal{C}}$ if and only if $C \in \mathcal{L}_n^{\mathcal{C}}$, and $CD \in \mathcal{L}_n^{\mathcal{C}}$ for any $C, D \in \mathcal{L}_n^{\mathcal{C}}$, an equivalence relation is implied.

The restriction to local Clifford operations might seem arbitrary at first. However, various methods and results concern LC-equivalence, some of which are presented in sec. 4.2 and sec. 4.4.1. The relation between LU-equivalence and LC-equivalence is discussed in sec. 4.4.2.

4.2 | Reduction to graph states

The second important result regarding the equivalence of stabilizer and graph states, is that every stabilizer state is LC-equivalent to at least one graph state. This was shown in [108] by making extensive use of the *binary representation* (see [93] or sec. 2.5). In this representation, Pauli operators are mapped to elements of $\mathbb{F}_2^{2n}$, so that a stabilizer $\mathcal{S}$ becomes an n-dimensional symplectic subspace spanned by the binary representations of its generators. *Symplectic* means that any element $x \in \mathbb{F}_2^{2n}$ of this subspace is self-orthogonal under a *symplectic inner product*:

$$x^T P x = 0, \tag{4.4}$$

where $P = \left[\begin{array}{c|c} 0 & \mathbb{I} \\ \hline \mathbb{I} & 0 \end{array}\right]$.

The Z- and X-supports are encoded into the first and last n bits of the vector, respectively, so that the stabilizer of an arbitrary stabilizer state is then represented by its *generator matrix* S:

$$S = \left[\begin{array}{c} Z \\ \hline X \end{array}\right], \tag{4.5}$$

where Z and X are matrices that encode the Z- and X-support of the generators of the stabilizer, so that S is full rank for the stabilizer of a stabilizer state. The stabilizer $\mathcal{S}$ is then represented by the symplectic subspace spanned by the columns of the generator matrix S, so it becomes somewhat independent of the specific basis. A change of basis for this subspace doesn't change the stabilizer, but represents a different set of generators for $\mathcal{S}$. Such a change of basis can be understood as an invertible matrix $R \in \mathbb{F}_2^{n\times n}$, so that two

generator matrices S and $S' = SR$ represent the same stabilizer and thus stabilizer state.

Due to the special structure of the generators of a graph state $|G\rangle$, its stabilizer $\mathcal{S}^{|G\rangle}$ has a generator matrix $S_{|G\rangle}$:

$$S_{|G\rangle} = \left[\begin{array}{c} \Gamma \\ \hline \mathbb{I} \end{array}\right], \tag{4.6}$$

where Γ is the adjacency matrix of the graph G.

A local Clifford operation on a stabilizer state is represented by an invertible matrix $Q \in \mathbb{F}_2^{2n\times 2n}$ that is in block diagonal form:

$$Q = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right], \tag{4.7}$$

i.e. A, B, C and D are all diagonal matrices. Additionally, Q must preserve the symplectic structure of the subspace of the generator matrix [109], which means that:

$$QPQ^{-1} = P. \tag{4.8}$$

It follows that a stabilizer state $|\psi\rangle$ with generator matrix S is local Clifford equivalent to a graph state $|G\rangle$ with generator matrix S_G, if and only if there exist invertible matrices Q (in the form of (4.7) and (4.8)) and $R \in \mathbb{F}_2^{n\times n}$ so that

$$S_G = QSR. \tag{4.9}$$

Careful inspection of the properties of the matrix S reveals that such Q and R always exist [108].

For a given stabilizer state $|\psi\rangle$, the graph state $|G\rangle$ to which it is local Clifford equivalent is not unique. Indeed, consider the graph G' that is obtained by a series of local complementations on G. Following sec. 3.3, the graph states $|G\rangle$ and $|G'\rangle$ are LC-equivalent, from which it follows that if $|\psi\rangle$ is LC-equivalent to $|G\rangle$, it is LC-equivalent to $|G'\rangle$ as well.

4.3 Orbits and entanglement classes

Considering the discussion of secs. 4.1 and 4.2, it can be very helpful to group together all graph states that are equivalent under a suitable set of operations. Following the discussion in sec. 4.1, the most general set for (connected) graph states of equal size is the set of local unitary operations. The set of all graph states $|G'\rangle$ that are LU-equivalent to a graph state $|G\rangle$ forms its *LU-orbit* $\mathcal{O}^{\mathrm{LU}}(|G\rangle)$.

Definition 20. *Let $|G\rangle$ be a graph state. The set of all graph states $|G'\rangle$ that are LU-equivalent to $|G\rangle$ is called the* LU-orbit $\mathcal{O}^{\mathrm{LU}}(|G\rangle)$ *of $|G\rangle$:*

$$\mathcal{O}^{\mathrm{LU}}(|G\rangle) = \{|G'\rangle \,|\, |G'\rangle \text{ is a graph state}, \exists U \in \mathcal{L}_n^{\mathcal{U}} \text{ s.t. } |G'\rangle = U|G\rangle\}. \tag{4.10}$$

Any element $|G'\rangle \in \mathcal{O}^{\mathrm{LU}}(|G\rangle)$ *is called a* representative *of the LU-orbit.*

Beyond LU-equivalence, it is also useful to restrict the set of allowed operations to only local Clifford operations, resulting in the *LC-orbit* $\mathcal{O}^{\mathrm{LC}}(|G\rangle)$.

Definition 21. *Let* $|G\rangle$ *be a graph state. The set of all graph states* $|G'\rangle$ *that are LC-equivalent to* $|G\rangle$ *is called the* LC-orbit $\mathcal{O}^{\mathrm{LC}}(|G\rangle)$ *of* $|G\rangle$*:*

$$\mathcal{O}^{\mathrm{LC}}(|G\rangle) = \{|G'\rangle \,|\, |G'\rangle \text{ is a graph state}, \exists C \in \mathcal{L}_n^{\mathcal{C}} \text{ s.t. } |G'\rangle = C\,|G\rangle\}. \quad (4.11)$$

Any element $|G'\rangle \in \mathcal{O}^{\mathrm{LC}}(|G\rangle)$ *is called a* representative *of the LC-orbit.*

There exists a very strong relation between the LC-orbit $\mathcal{O}^{\mathrm{LC}}(|G\rangle)$ of a graph state $|G\rangle$ and the (local-complementation) orbit $\mathcal{O}(G)$ of the associated graph G (see sec. 3.1.1)[2]. This relation is discussed in sec. 4.4.1.

Entanglement classes

Historically, two graph states that are the same up to a permutation of their qubits were considered to have identical entanglement. More specifically, usually *unlabelled* graphs were considered, so that the nodes of the graph have no ordering, and there is no notion of permutation. To make the distinction with labelled graphs more precise, it is easier to still assume labelled graphs, but consider permutations of the nodes of a graph to result in 'equivalent' graphs. More specifically, let $\mathcal{V}_n$ be the permutation group of n elements, and let $\sigma \in \mathcal{V}_n$ be any permutation. The notation $\sigma(G)$ then indicates the (labelled) graph that results from permuting the nodes of the graph G with σ. The resulting graph $G' = \sigma(G)$ is called a *permutation* of G. For a given graph state $|G\rangle$, the set of all graph states $|G'\rangle$ that are LU-equivalent to $|G\rangle$, or where there exist a permutation $\sigma \in \mathcal{V}_n$ such that $|G'\rangle = |\sigma(G)\rangle$, or both at the same time, is then called the *entanglement class* $\mathcal{E}_{\mathcal{C}}(|G\rangle)$ of $|G\rangle$.

Definition 22. *Let* $|G\rangle$ *be a graph state and let* $\mathcal{O}^{\mathrm{LU}}(|G\rangle)$ *be its LU-orbit. The* entanglement class $\mathcal{E}_{\mathcal{C}}(|G\rangle)$ *of* $|G\rangle$ *is the set of all graph states* $|G'\rangle$*, for which there exists a permutation* $\sigma \in \mathcal{V}_n$ *such that* $|\sigma(G')\rangle$ *is in* $\mathcal{O}^{\mathrm{LU}}(|G\rangle)$*:*

$$\mathcal{E}_{\mathcal{C}}(|G\rangle) = \{|G'\rangle \,|\exists \sigma \in \mathcal{V}_n : |\sigma(G')\rangle \in \mathcal{O}^{\mathrm{LU}}(|G\rangle)\}. \quad (4.12)$$

Any element $|G'\rangle \in \mathcal{E}_{\mathcal{C}}(|G\rangle)$ *is called a* representative *of the entanglement class.*

Because $\mathcal{L}_n^{\mathcal{C}} \subset \mathcal{L}_n^{\mathcal{U}}$, the following inclusion relation follows:

$$\mathcal{O}^{\mathrm{LC}}(|G\rangle) \subseteq \mathcal{O}^{\mathrm{LU}}(|G\rangle) \subseteq \mathcal{E}_{\mathcal{C}}(|G\rangle). \quad (4.13)$$

[2]Note that, to ease the distinction, the parameter of an LU- or LC-orbit will always be a graph state $|G\rangle$. Thus, it will always be written as $\mathcal{O}^{\mathrm{LU}}(|G\rangle)$ or $\mathcal{O}^{\mathrm{LC}}(|G\rangle)$, but never written as $\mathcal{O}^{\mathrm{LU}}(G)$ or $\mathcal{O}^{\mathrm{LC}}(G)$. The *local complementation* orbit of a graph G is then written $\mathcal{O}(G)$.

Any entanglement class $\mathcal{E}_{\mathcal{C}}(|G\rangle)$ can be 'built' as the aggregate of all the different LU-orbits that are associated with it; these are exactly the LU-orbit $\mathcal{O}^{\mathrm{LU}}(|G\rangle)$, and all its permutations. This means that multiple, distinct LU-orbits are associated with every entanglement class, but every unique permutation of a graph does not necessarily create a new LU-orbit associated with the entanglement class. Indeed, usually there are permutations of a graph that leave it invariant. Moreover, there can be multiple permutations σ of a graph G so that the resulting graph $\sigma(G)$ is not identical to G, but its graph state $|\sigma(G)\rangle$ is LU-equivalent to $|G\rangle$. It follows that not every permutation $\sigma \in \mathcal{V}_n$ necessarily gives its own distinct LU-orbit, so that the total number of distinct LU-orbits associated with an entanglement class is generally smaller than the total number of permutations, $|\mathcal{V}_n| = n!$.

An instructive example is given by the entanglement class $\mathcal{E}_{\mathcal{C}}(|L_{1234}\rangle)$ of the four-qubit linear cluster state $|L_{1234}\rangle$, shown in **Fig.** 4.1. The LU-orbit $\mathcal{O}^{\mathrm{LU}}(|L_{1234}\rangle)$ of $|L_{1234}\rangle$ consists of 11 elements, which are shown in the first row of **Fig.** 4.1. The states $|L_{1234}\rangle$, $|L_{2134}\rangle$, $|L_{2143}\rangle$ and $|L_{1342}\rangle$ (highlighted in yellow), are all elements of $\mathcal{O}^{\mathrm{LU}}(|L_{1234}\rangle)$ that follow from permutations of L_{1234}.

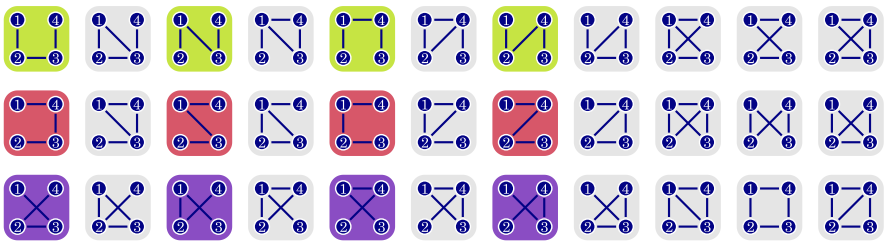


Figure 4.1: The entire entanglement class $\mathcal{E}_{\mathcal{C}}(|L_{1234}\rangle)$ of the state $|L_4\rangle = |L_{1234}\rangle$. It consists of three different LU-orbits, that are each shown in a separate row. For each row, the permutations of the first graph that remain in the same LU-orbit are highlighted. The top row is $\mathcal{O}^{\mathrm{LU}}(|L_{1234}\rangle)$, the LU-orbit of $|L_{1234}\rangle$, the middle row is the LU-orbit $\mathcal{O}^{\mathrm{LU}}(|L_{1432}\rangle)$, and the bottom row is the LU-orbit $\mathcal{O}^{\mathrm{LU}}(|L_{1324}\rangle)$.

The number of LU-orbits associated with an entanglement class can be determined by inspection of the permutations. Let $\mathcal{D}$ be the set of all permutations of G so that the associated graph state falls into the LU-orbit of $|G\rangle$:

$$\mathcal{D} = \{\sigma \in \mathcal{V}_n | \, |\sigma(G)\rangle \in \mathcal{O}^{\mathrm{LU}}(|G\rangle)\}. \tag{4.14}$$

This set forms a subgroup of the permutation group $\mathcal{V}_n$, and all of the different LU-orbits associated with the entanglement class of $|G\rangle$ are represented by the different cosets of this subgroup. It follows from Lagrange's theorem that the number of LU-orbits per entanglement class is $\frac{n!}{|\mathcal{D}|}$.

For the nodes of $|L_{1234}\rangle$ in **Fig.** 4.1, there are $4! = 24$ permutations in total. As noted before, the four permutations that give L_{1234}, L_{2134}, L_{2143} and L_{1342}, as well as their reversals, are all part of the set $\mathcal{D}$, for a total of eight elements. It follows that there are $\frac{24}{8} = 3$ distinct LU-orbits associated with the entanglement class $\mathcal{E}_\mathcal{C}(|L_{1234}\rangle)$. The other two LU-orbits, shown in **Fig.** 4.1 as the other two rows, can be interpreted as $\mathcal{O}^{\mathrm{LU}}(|L_{1432}\rangle)$ and $\mathcal{O}^{\mathrm{LU}}(|L_{1324}\rangle)$, so that:

$$\mathcal{E}_\mathcal{C}(|L_{1234}\rangle) = \mathcal{O}^{\mathrm{LU}}(|L_{1234}\rangle) \cup \mathcal{O}^{\mathrm{LU}}(|L_{1432}\rangle) \cup \mathcal{O}^{\mathrm{LU}}(|L_{1324}\rangle). \tag{4.15}$$

Note that the LU-orbit $\mathcal{O}^{\mathrm{LU}}(|L_{1234}\rangle)$, i.e. the first row of **Fig.** 4.1, consists exactly of the graphs in the local complementation orbit of L_{1234} (see **Fig.** 3.3). This is no coincidence but exemplary of a broader fact that will be discussed in sec. 4.4.1.

Another instructive example of an entanglement class is given by the GHZ state (see Def. 17). From **Fig.** 3.5 it is straightforward to see that any permutation of the star graph either results in the same star graph, or in a star graph with another central node. The other star graphs are related to the original star graph by local complementations, so the associated graph states are LC-equivalent (see sec. 3.3). It follows that the LU-orbit and entanglement class of the GHZ state are identical, $\mathcal{O}^{\mathrm{LU}}(|\mathrm{GHZ}_n\rangle) = \mathcal{E}_\mathcal{C}(|\mathrm{GHZ}\rangle)$.

The total number of LU-orbits and entanglement classes grows quickly with the number of qubits. Similarly, the size (i.e. the number of elements) of a single LU-orbit or entanglement class generally grows quickly with the number of qubits. **Tab.** 4.1 details the number, and average and maximum sizes of all LU-orbits and entanglement classes up to nine qubits. Note that the number of LU-orbits times its average size is equal to the number of entanglement classes times *its* average size, and is exactly the total number of connected (labelled) graphs of a given size[3].

Finally, it is useful to extend LC-orbits to additionally include permutations, similar to how LU-orbits and entanglement classes are related. This results in an *LC-class*[4].

Definition 23. *Let $|G\rangle$ be a graph state and let $\mathcal{O}^{\mathrm{LC}}(|G\rangle)$ be its LC-orbit. The* LC-class *$\mathcal{E}^{\mathrm{LC}}_\mathcal{C}(|G\rangle)$ of $|G\rangle$ is the set of all graph states $|G'\rangle$, for which there exists a permutation $\sigma \in \mathcal{V}_n$ such that $|\sigma(G')\rangle$ is in $\mathcal{O}^{\mathrm{LC}}(|G\rangle)$:*

$$\mathcal{E}^{\mathrm{LC}}_\mathcal{C}(|G\rangle) = \{|G'\rangle \,|\exists \sigma \in \mathcal{V}_n : |\sigma(G')\rangle \in \mathcal{O}^{\mathrm{LC}}(|G\rangle)\}. \tag{4.16}$$

Any element of $|G'\rangle \in \mathcal{E}^{\mathrm{LC}}_\mathcal{C}(|G\rangle)$ is called a representative *of the LC-class.*

Similarly to (4.13), it holds that:

$$\mathcal{O}^{\mathrm{LC}}(|G\rangle) \subseteq \mathcal{E}^{\mathrm{LC}}_\mathcal{C}(|G\rangle) \subseteq \mathcal{E}_\mathcal{C}(|G\rangle). \tag{4.17}$$

[3] The number of connected labelled graphs of size n can be retrieved on the OEIS, the *online encyclopedia of integer sequences*, specifically sequence nr. `A001187` [110].

[4] This is not a term found in literature. Sometimes in literature it is called *non-isomorphic LC-orbit*, but I find this vague and ambiguous.

n	3	4	5	6	7	8	9
#	1	4	27	312	6103	214722	14639499
aver.	4.0	9.5	27.0	85.6	305.8	1171.5	4528.6
max.	4	11	132	372	1096	3248	9432
#	1	2	4	11	26	101	440
aver.	4.0	19.0	182.0	2427.6	71779.1	2490580.1	150673388.8
max	4	33	450	7920	378720	30280320	3206407680

Table 4.1: The number, and average- and maximum sizes of all LU-orbits $\mathcal{O}^{\mathrm{LU}}(|G\rangle)$ (middle rows) and entanglement classes $\mathcal{E}_{\mathcal{C}}(|G\rangle)$ (bottom rows) for size $3 \leqslant n \leqslant 9$. There are many more LU-orbits than entanglement classes, because with every entanglement class there are $\mathcal{O}(n!)$ different LU-orbits associated: permutations of graphs that are not associated with the LU-orbit, can create a new LU-orbit associated with the entanglement class.

4.4 | Local equivalence of graph states

This section addresses the local equivalence of graph states, for which it suffices to consider only local unitary operations, following the discussion in sec. 4.1. More specifically, this section aims to provide tools to determine if, for two graph states $|\psi\rangle_G$ and $|\psi\rangle_H$, it holds that $|\psi\rangle_G \in \mathcal{O}^{\mathrm{LU}}(|\psi\rangle_H)$. However, it is useful to first consider only local Clifford operations, which is done in sec. 4.4.1. In sec. 4.4.2, the extension to local unitary equivalence is made, and the difference between the LC-orbit $\mathcal{O}^{\mathrm{LC}}(|G\rangle)$ and the LU-orbit $\mathcal{O}^{\mathrm{LC}}(|G\rangle)$ of a graph state $|G\rangle$ is discussed.

4.4.1 | Local Clifford equivalence of graph states

In sec. 3.3 it was shown that the graph states $|G\rangle$ and $|\tau_i(G)\rangle$ are related by a local Clifford operation, for a local complementation τ_i on any node i. Because the local Clifford operators form a group, it follows that *any* graph $H \sim G$ (i.e. $H \in \mathcal{O}(G)$, see sec. 3.1), gives a graph state $|H\rangle$ that is LC-equivalent to $|G\rangle$. In other words, it holds that:

$$H \in \mathcal{O}(G) \Rightarrow |H\rangle \in \mathcal{O}^{\mathrm{LC}}(|G\rangle), \tag{4.18}$$

i.e. any graph in the orbit of G gives a graph state that is LC-equivalent to $|G\rangle$.

The third important result that was mentioned in the introduction is that the reverse of (4.18) is true as well, which was proven in [108]. If a graph state $|H\rangle$ is LC-equivalent to a graph state $|G\rangle$, then it holds that $H \sim G$, i.e. they are related by a series of local complementations. So, it holds that

the LC-orbit of a graph state $|G\rangle$ and the set of graph states associated with the orbit of G are the same:

$$\mathcal{O}^{\mathrm{LC}}(|G\rangle) = \{|H\rangle \, | H \in \mathcal{O}(G)\} \, . \tag{4.19}$$

Determining if two graph states are LC-equivalent is therefore equivalent to determining if the two associated graphs are in the same orbit.

To do so, it can be helpful to use the binary representation of the stabilizers of the two graph states. Specifically, let the two stabilizers have generator matrices S_G and S_H, respectively. If the two graph states are LC-equivalent, there exists an invertible matrix Q of the form (4.7) so that S_G and QS_H represent the same stabilizer. The stabilizer is represented by the (self-orthogonal) subspace spanned by the columns of the generator matrix, so it follows that S_G and QS_H represent the same stabilizer if and only if it holds that:

$$S_G^T P Q S_H = 0, \tag{4.20}$$

for some Q in the form of (4.7) that is additionally subject to the condition of (4.8). Compare this with (4.9): although that equation gives a closed form for the graph state that is local equivalent to the state $|\psi\rangle$ with generator matrix S, (4.20) is easier to use as a test for equivalence, because the change-of-generators matrix R does not have to be specified.

If Γ_G and Γ_H are the adjacency matrices of G and H, respectively, (4.20) reduces to (using (4.6) and (4.7)):

$$\Gamma_G B \Gamma_H \oplus D\Gamma_H \oplus \Gamma_G A \oplus C = 0, \tag{4.21}$$

where '$\oplus$' indicates entry-wise addition over $\mathbb{F}_2$. This results in a set of linear equations of $4n$ variables, which can be solved by Gaussian elimination in $\mathcal{O}(n^4)$ steps. Any element of the set of solution of this system, which is essentially the kernel of (4.21), additionally needs to be subject to (4.8) to be a true solution. That is, every element in the set of solutions needs to be checked against this condition. Note that this set forms a subspace of some dimension $d \leqslant 4n$, so that there are 2^d elements to check. This means that, in the worst case, the condition would have to be checked against an exponential number of elements.

The fourth important result mentioned in the introduction is that the set of solutions for which (4.8) needs to be checked, can be reduced to a smaller set that is polynomial in size. Specifically, let $\{b_i\}_{i=1}^d$ be a basis of the subspace of solutions to (4.21), and suppose it has more than four elements[5]. Then it suffices to check (4.8) only for the elements $\{b_i \oplus b_j\}_{i \neq j}$ of the subspace, i.e. the sums of pairs of distinct basis elements. This gives $\mathcal{O}(d^2)$ elements to check, instead of all 2^d linear combinations of the basis elements. This was

[5] If there are only four basis vectors, all $2^4 = 16$ elements need to be checked, but this is manageable.

originally proven and presented in [111, 112] in purely graph theoretic terms, where it was presented as an algorithm to verify if two graphs G and H are in each others' orbit. It was then introduced to the quantum community and connected to graph states in [113], and the formulation of that publication has been used here. Nevertheless, the method explained here, in the form of an algorithm, is known as the *Bouchet algorithm*, named after the author of the original presentation [111].

Although the Bouchet algorithm gives an efficient method to determine the LC-equivalence of two graph states, many other properties are hard to determine. Calculating the size $\left|\mathcal{O}^{\mathrm{LC}}(|G\rangle)\right|$ of the LC-orbit of a random graph state $|G\rangle$ is $\#\mathbb{P}$-complete [114], which means that it is at least as hard as an $\mathbb{NP}$-complete problem. Separately, the element of $\mathcal{O}(G)$ with the least number of edges has an important operational meaning, because it corresponds to the element of $\mathcal{O}^{\mathrm{LC}}(|G\rangle)$ that takes the smallest number of C_Z gates to prepare. Determining this element is essentially the same as calculating every element of the orbit, which is computationally hard [114, 115].

4.4.2 | Local unitary versus local Clifford equivalence of graph states

The results of sec. 4.4.1 give a clear graphical tool to understand the LC-equivalence of graph states, and a programmatic method to efficiently determine such LC-equivalence. The methods form important tools in the study of LU-equivalence as well, because any two graph states that are LC-equivalent, are by definition LU-equivalent as well.

It was a long-standing conjecture that the converse also holds, implying that LU-equivalence and LC-equivalence of graph states are identical. It was shown that this conjecture, known as the *LU-LC-conjecture*, holds for up to 8 qubits [116], but it should be noted that this was only done for *unlabelled* graphs. This means that for every graph state $|G\rangle$ with eight or fewer qubits it holds that $\mathcal{E}_{\mathcal{C}}(|G\rangle) = \mathcal{E}_{\mathcal{C}}^{\mathrm{LC}}(|G\rangle)$.

Recent work (Pubs. **[G]** and **[H]** ([55, 117])) has increased this threshold to 11 qubits. Additionally, the same work has shown that for *labelled* graphs the conjecture holds up to at least 8 qubits, so that $\mathcal{O}^{\mathrm{LU}}(|G\rangle) = \mathcal{O}^{\mathrm{LC}}(|G\rangle)$. These results will be discussed in chapter 6.

For graph states of any size it was shown [118] that the conjecture must hold for specific subclasses of all graph states; these subclasses make use of the marginal properties of the graph states (discussed in chapter 6), and include those graphs for which:

$$\Gamma^2 \oplus \Gamma = \mathbb{I}. \tag{4.22}$$

The set of graph states for which the LU-LC-conjecture holds was extended to include all graphs without cycles of either three or four nodes [119].

However, [120] ultimately showed that in general the conjecture is false, using a counterexample of 27 qubits. This was followed by a constructive family of counterexamples of at least 27 qubits [51], for which the previous

counterexample is the smallest member. It is an open question what the smallest counterexample to the LU-LC conjecture is.

4.5 | Equivalence involving measurements

As shown in section sec. 3.4, single qubit measurements on graph states result in post-measurement states that are themselves graph states, up to local Clifford corrections. These correction operators can depend on the measurement outcomes, so in a networked setting the situation fits neatly into the SLOCC paradigm. Given the graphical rules explained in that same section, it follows that a graph state $|G_H\rangle$ can be obtained from another graph state $|G_G\rangle$ by local Clifford operations and Pauli measurements if and only if H can be obtained from G by local complementations and node deletions. This statement is made more precise in [52], using the concept of *vertex minors* [121, 122].

Definition 24. [52] *Let G and H be two graphs such that $V(G) \subseteq V(H)$ (see Def. 13). Then G is a* vertex minor *of H, denoted $G < H$, if G can be obtained from H by a series of local complementations and node deletions on H.*

The definition of vertex minors is useful, because for two random graphs G and H, the associated graph state $|G\rangle$ can be obtained from the graph state $|H\rangle$ by local Clifford operations, single-qubit Pauli measurements and classical communication if and only if $G < H$ [52]. As explained in the introduction, $|G\rangle$ and $|H\rangle$ are referred to as the *target* and *resource* graph states, respectively.

When $G < H$ the local complementations and node deletions that relate H to G can be understood to occur in arbitrary order. Still, it was shown in [52] that equivalently one can assume all local complementations to occur together, before any nodes are deleted. This means that if $G < H$, there exists some graph $H' \in \mathcal{O}(H)$ such that $G = H' \setminus m$, where $m = \{V(H) \setminus V(G)\}$, i.e. the nodes that are in H but not in G. Even though this gives a simpler condition, it is still $\mathbb{NP}$-complete to determine if $G < H$ for two randomly chosen graphs G and H [123].

If there is structure in either the resource or target graph states, this can potentially be used to efficiently determine if the associated graphs are vertex minors. If the *rank-width* [122] of the graph associated with the resource state is bounded, it is efficient to decide if a target graph state can be obtained [123] (using results from [122]). This rank-width is a complexity measure of a graph, and is strongly related to the entanglement entropy (see Def. 6) of the associated graph state [124]. It should be noted that, even though it is technically efficient to decide if $G < H$ for fixed rank-width, the coefficients

in the polynomial scaling are exceedingly[6] large [123]. However, for specific choices of resource or graph states it is practically possible to determine if one can be obtained from the other. Chapter 5 presents such a specific choice.

4.6 Conclusion

The results and methods introduced in this chapter offer strong tools to study the entanglement of stabilizer states. However, especially for the case where measurements are involved, the general questions are still hard to answer. As mentioned in sec. 4.5, if there is usable structure in either of the states, it can sometimes be determined if a target state can be obtained from a resource state.

Chapter 5 presents the contents of Pub. **[F]** ([68]), which takes the specific choice of the linear cluster state and the GHZ state as the resource and target state, respectively. It gives a full characterization of which choices of node deletions are possible, and which are not.

The discussion in sec. 4.5 only considers single qubit measurements in the Pauli-bases, and local Clifford operations. From sec. 4.4.2 it follows that local Clifford operations do not give the most general setting, and a similar argument can be made regarding measurements in only Pauli-bases. Generalising to local unitary operations and non-Pauli-basis measurements could potentially widen the set of target states that can be obtained from a given resource state, but this general question has not been broadly studied and seems to be hard to answer.

Using the Bouchet algorithm to determine LU equivalence is not infallible, because it only checks for LC-equivalence. However, it can still provide conclusive results if the states *are* LC-equivalent, because this implies LU-equivalence. Moreover, it is highly unlikely that the algorithm gives a false negative, because the LU-LC conjecture holds for a vast majority of graphs. Nevertheless, it is technically possible that wrong results are obtained, as shown by the counterexamples in [51, 120]. An algorithm to verify LU-equivalence of general n-qubit states was presented in [125, 126], but this algorithm is generally inefficient, because the number of steps to check for equivalence is exponential in n.

Moreover, these algorithms determine the equivalence of (graph) states by direct pairwise comparison. This means that to determine the equivalence of a set of L graph states, the algorithm has to be run a total of $\frac{L(L-1)}{2}$ times. Chapter 6 introduces methods from Pub. **[G]** ([55]) that can be used to *categorize* the LU-orbit or entanglement class from a single graph state.

[6] This coefficient is a 'power tower' of ten 2's followed by the rank width r, and is so unfathomably large that it is not even possible to practically write it down in scientific notation.

Recent work, Pub. **[H]** ([117]), introduced a novel algorithm to determine the LU-equivalence of graph states, which is separate from the Bouchet algorithm, but this publication is not addressed in this thesis.

5

Extracting GHZ states by local operations

Reach for the stars!

Buzz Lightyear, *Toy Story (Pixar)*

As discussed in the conclusion of chapter 4, it is hard in general to decide if one graph state can be obtained from another by local operations and single-qubit measurements. When only Pauli measurements and local Clifford operations are considered, determining if the *target* state $|G_t\rangle$ can be obtained from the *resource* state $|G_r\rangle$ reduces to determining if $G_t < G_r$, i.e. if the associated graphs are related by a vertex-minor relation (see Def. 24). Still, the problem is $\mathbb{NP}$-complete in general [123], but it might be possible to answer it when there is structure in either or both of the resource and target graph states that can be utilized. When $|G_t\rangle$ can indeed be obtained from $|G_r\rangle$, it is said that the target state can be *extracted* from the resource state.

This chapter presents the contents of Pub. **[F]** ([68]), where specific choices for both the resource and the target graph state are made. The resource graph state is the n-qubit *linear cluster state* (see Def. 16), and the target graph state is the m-qubit GHZ state (see Def. 17). When $m = n > 3$ (for the case m = n = 3, see **Fig.** 3.6), the LU-orbits $\mathcal{O}^{\mathrm{LU}}(|\mathrm{GHZ}_m\rangle)$ and $\mathcal{O}^{\mathrm{LU}}(|L_n\rangle)$ of the GHZ and linear cluster state are not the same, which means that the states

are not LU-equivalent. It follows that at least one node of the resource state has to be measured, so that $m < n$.

Because there is at least one node that is measured, the target graph state is extracted on a specific choice of the original n qubits. This selection can play a significant role, meaning that only for specific selections of the original qubits extraction is possible. A specific selection of nodes on which the GHZ state is to be realized is referred to as an *extraction pattern*, or just a *pattern* if context allows. Following the same terminology, an extraction pattern is *possible* if the target graph state can be extracted from the resource state, or *impossible* if this is not the case. Finally, note that the local unitary operations are allowed to depend on outcomes of measurements on other nodes, so that the situation fits into the LOCC paradigm (see sec. 4.1).

Extraction is possible only for specific patterns; this chapter gives a complete characterization of which patterns are possible, and which are not. Any extraction pattern with a certain structure in the choice of nodes will be shown to be impossible, following a theorem that was originally presented in Pub. **[F]**. Any other pattern is possible, which is shown by explicit construction. Hence, a complete characterization is obtained for the extraction of GHZ states from linear cluster states, from which it follows that there is an upper bound on m in terms of n, so that no GHZ that is larger can be obtained. To complement the theoretical analysis, Pub. **[F]** presented the results of an implementation of the extraction on real hardware.

The post-measurement state that is obtained from the linear cluster state after performing the extraction, is LC-equivalent to the GHZ state. In general, the local Clifford operations to obtain the true target state depend both on the specific extraction pattern and on the outcomes of the measurements. Certain details regarding these *correction operators* were not presented in Pub. **[F]** but deferred to its supplementary material Sup. **[sB]** ([127]). In this thesis, the same details have been deferred to chapter B.

First, sec. 5.1 introduces some useful terminology and makes the setting more precise. In sec. 5.2 a useful theorem is presented, that can be used to show the impossibility of many extraction patterns. The same theorem and associated corollaries help to bound m in terms of n without having to resort to specific extraction patterns. A specific extraction pattern that saturates this bound, referred to as the *maximal pattern*, is introduced in sec. 5.3. In sec. 5.4 any other extraction pattern that is not strictly ruled out by the previous results, is shown to be possible by a constructive argument. The experimental implementation is presented in sec. 5.5, and the chapter is concluded in sec. 5.6.

5.1 Setting

To ease the discussion, useful terminology and notation is introduced. The set $V_L = [n] = \{1, \ldots, n\}$, referred to as the *network*, is a set of n qubits. The

linear cluster state $|L_{V_L}\rangle$ defined on the network V_L is the resource state. The goal is to extract a $|\mathrm{GHZ}_{V_G}\rangle$ state on a subset $V_G \subset V_L$, referred to as the *extraction pattern*, which has $m = |V_G|$ qubits. This chapter will detail which choices of V_G are possible, and which are not.

It is useful to represent the nodes of V_L as lying on a horizontal line, which is naturally implied by the linear structure of the resource state. Specifically, this means that node i is adjacent to nodes $i-1$ and $i+1$; these nodes are referred to as the left- and right *neighbours*, respectively, of node i. If a node has no neighbours on the left or on the right (i.e. for nodes 1 and n), that node is said to be on the left and right *edge*, respectively. Moreover, nodes can be *left* or *right* of each other, even if there exist other nodes between them, and concepts like *consecutive* nodes apply as well.

This terminology can be adopted to refer to the nodes in the extraction pattern V_G, so that they can be referred to be e.g. *left* of other nodes in the pattern. Moreover, the *boundaries* of the extraction pattern can be used to refer to the leftmost- and rightmost node within the pattern.

Additionally, it is useful to refer to a set of consecutive nodes in the extraction pattern with the term *island*.

Definition 25. *A k-island is a series of k consecutive nodes $\{i, \ldots, i+k-1\} \in V_L$ (for some i) that are all part of the extraction pattern, so that $\{i, \ldots, i+k-1\} \in V_G$. A k-island at node i in V_G indicates the set $\{i, \ldots, i+k-1\}$, i.e. the k-island with its leftmost node positioned at node i.*

Note that, as defined, a subset of an island can be an island itself; if V_G would contain e.g. the 3-island $\{1,2,3\}$, then $\{1,2\}$ and $\{2,3\}$ would be a 2-islands of V_G. k-islands play an important role in determining which extraction patterns are possible, and which ones are not.

5.2 | Impossible extraction patterns

There are strong restrictions imposed on the possibilities of extraction patterns containing islands, which the following theorem shows.

Theorem 1. (Pub. **[F]**) *No 2-island can have both a left and a right neighbour in V_G. This means that if there is a 2-island at node i, there is no node left of i or right of $i+1$ in V_G.*

In other words, if the extraction pattern contains a 2-island, it is necessarily at the boundary of the extraction pattern. Thm. 1 leads directly to the fact that all nodes in V_G are 'isolated': if node i is in the extraction pattern, then neither $i-1$ nor $i+1$ can be in the extraction pattern, except for the two boundaries. The proof of Thm. 1 is deferred to chapter A.

A useful corollary to Thm. 1 that concerns larger 2-islands follows.

Corollary 1. (Pub. **[F]**) *If V_G contains a 3-island, then $|V_G| = 3$, i.e. they are the only nodes in the pattern.*

Proof. The proof is by straightforward contradiction. Assume that V_G contains the 3-island $\{i, i+1, i+2\}$ and additionally some other node j. W.l.o.g. assume that $j > i+2$, i.e. it is to the right of the 3-island. Then, $\{i+1, i+2\}$ is a 2-island with both a left neighbour (node i) and a right neighbour (node j) in V_G, which is in direct contradiction to Thm. 1. □

Fig. 5.1 contains three examples of (possible or impossible) extraction patterns. The resource state is $|L_8\rangle$, and the nodes in the extraction pattern have been highlighted. Two of the three given examples are prohibited by Thm. 1 and have been marked by ✗. The other example, marked by ✓, is not directly prohibited by Thm. 1. That this extraction pattern is indeed possible does not follow from Thm. 1, but it is proven in sec. 5.4.

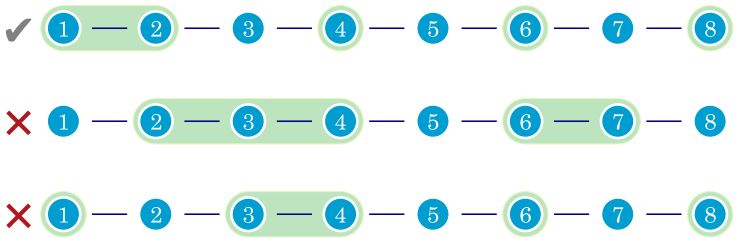


Figure 5.1: Three different choices for the *extraction patterns* V_G are depicted, on which the $|\mathrm{GHZ}_{V_G}\rangle$ state is to be extracted from the $|L_8\rangle$ state. The highlighted nodes are those nodes in V_G; the other nodes are measured during the extraction. For every pattern it is shown by ✗ or ✓ if extraction is made impossible or not by Thm. 1. Note that Thm. 1 only prohibits certain extraction patterns, but does not say anything about the possibility of patterns that it does not prohibit. In secs. 5.3 and 5.4 it is shown that all other patterns are possible.

Another corollary to Thm. 1 gives an upper bound to the size of any GHZ state that can be extracted from a linear cluster state.

Corollary 2. (Pub. **[F]**) *Let V_G be a pattern to extract a* GHZ *state from a linear cluster state of size n. Then, the size of V_G is bounded from above:*

$$|V_G| \leqslant \left\lfloor \frac{n+3}{2} \right\rfloor . \tag{5.1}$$

Proof. The proof is by simple arithmetic. There can be at most two 2-islands by Thm. 1, so the number of nodes in V_G is maximized by including the left 2-island $\{1, 2\}$, and the right 2-island $\{n-1, n\}$. To not violate Thm. 1,

every other alternating node has to be measured, i.e. the nodes $3, 5, \cdots \notin V_G$. For odd n, this sequence ends at node $n-2$, so that $k = {}^{n-3}/_{2}$ nodes have been measured. For even n, this sequence would end at $n-3$, inadvertently creating the 3-island $\{n-2, n-1, n\}$. So, node $n-2$ is also measured, for $k = {}^{n-2}/_{2}$ nodes in total. In both cases it holds that $|V_G| = n - k$, and the bound follows. □

Note that the additional node that is removed in the proof of Cor. 2 when n is even, does not have to be $n-2$. It can be either 1 or n, or any other internal node so that two other consecutive nodes are measured.

An extraction pattern of a similar but slightly smaller size $|V_G| = {}^{n}/_{2}$ was reported in [128], albeit without proof or claims of maximality. In the same publication the concept of *entanglement persistency* is introduced; the entanglement persistency of a (multi-qubit) state is equal to the minimum number of single-qubit measurements that need to be performed, so that the post-measurement state is completely separable. This concept was used to prove Cor. 2 in [129], but cannot be used to prove either Thm. 1 or Cor. 1.

It remains to be proven that the pattern given in the proof to Cor. 2 is indeed a possible pattern. The pattern of the proof of Cor. 2 for odd n, referred to as the *maximal extraction pattern*, is discussed and proven possible in sec. 5.3. Subsequently, in sec. 5.4 it is shown that any other pattern not prohibited by Thm. 1 is possible as well, by reducing it to the maximal extraction pattern.

5.3 The maximal extraction pattern

The maximal extraction pattern $V_G = \{1, 2, 4, 6, \ldots, n-3, n-1, n\}$ dictates that the nodes $j \in \{3, 5, \ldots, n-2\}$ are measured, for a total of ${}^{(n-3)}/_{2}$ measurements. Indeed, the post-measurement state is (LC-equivalent) to the star graph state if all of these nodes are measured in the X basis, which means that the set of measurement operators for the extraction is $\{X_j\}_{j=3,5,\ldots,n-2}$.

An X-basis measurement can be interpreted (see sec. 3.4.3) as a Y-basis measurement, preceded by a local complementation on a random neighbour of the measured node[1]. Measuring the nodes in ascending order, taking the right neighbour of every measured node to perform the local complementation, results in the star graph centred around node 2, from which it follows that the post-measurement state is LC-equivalent to the GHZ state (see Def. 17). **Fig.** 5.2 shows the pattern for seven qubits, which generalises to any (odd) n.

[1] As explained in sec. 3.4.3, sometimes an additional local complementation on the random neighbour is included after the node deletion. However, this does not affect the current discussion, as can be easily seen in **Fig.** 5.2: the chosen neighbours are *leaves* [130] and are therefore unaffected by a local complementation

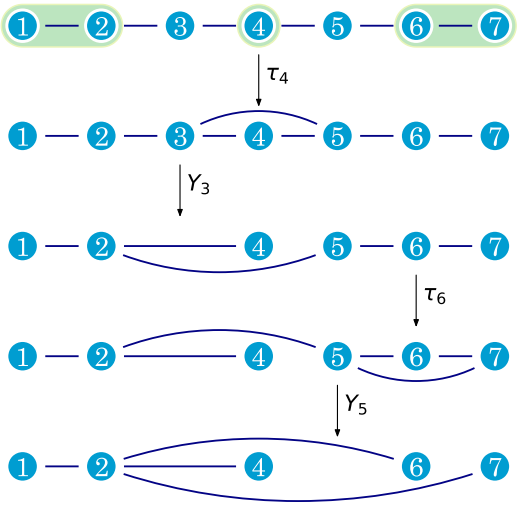


Figure 5.2: GHZ state extraction from $|L_7\rangle$ with the extraction pattern $V_G = \{1, 2, 4, 6, 7\}$ (highlighted in the top row), which is the maximal extraction pattern for $n = 7$. All nodes not in the extraction pattern are measured in the X basis, which can be viewed as a Y-basis measurement of the node preceded by a local complementation on a random neighbour (see sec. 3.4.3). Here, the right neighbour is chosen as the random neighbour, so that the consecutive measurements result in the star graph centred at node 2. Following Def. 17, this is LC-equivalent to the GHZ state, so that the extraction is indeed possible. This method generalizes straightforwardly to linear cluster states of arbitrary (odd) size.

To determine the exact local operations necessary to obtain the GHZ state, a more careful analysis is needed, which can be obtained by inspection of the generators of the linear cluster state. **Tab.** 5.1 contains these generators, grouped by odd (top) and even (bottom) index. Additionally, the generator g_2 is changed to $g_2' = \prod_{\{\text{even } i\}} g_i = g_2 g_4 g_6 \ldots$, so that the measurement operators $\{X_j\}_{j=3,5,\ldots,n-2}$ commute with all odd-indexed generators, and with g_2'. The other $|V_L| - |V_G| = {}^{(n-3)}/_2$ generators, $g_4, g_6, \ldots, g_{n-1}$, all anti-commute with at least one measurement operator X_j. Note that there is an equal number of anti-commuting generators and measurement operators. Following the discussion in sec. 2.3, the measurement can be interpreted as the measurement operators replacing the anti-commuting generators, provided they carry the measurement outcomes $m_j = \pm 1$ as the phase. The post-measurement state is then generated by the odd-indexed original generators,

the generator g_2', and the measurement operators $(m_j)X_j$:

$$\begin{aligned} & g_i && (i \text{ odd}), \\ & g_2', && \\ & (m_j)X_j && (j = 3, 5, \ldots, n-2). \end{aligned} \tag{5.2}$$

	1	2	3	4	5	6	$\cdots$	$n-3$	$n-2$	$n-1$	n
g_1	X	Z					$\ldots$				
g_3		Z	X	Z			$\ldots$				
g_5				Z	X	Z	$\ldots$				
$\ldots$							$\ldots$				
g_{n-2}							$\ldots$	Z	X	Z	
g_n							$\ldots$			Z	X
g_2'	Z	X		X			$\ldots$	X		X	Z
g_4			Z	X	Z		$\ldots$				
g_6					Z	X	$\ldots$				
$\ldots$							$\ldots$				
g_{n-3}							$\ldots$	X	Z		
g_{n-1}							$\ldots$		Z	X	Z

Table 5.1: Generators of the odd-n $|L_n\rangle$ state (see Def. 16). The generators with odd and even indices have been grouped separately, and $g_2 = Z_1X_2Z_3$ has been changed to $g_2' = g_2g_4g_6\ldots$. Now only the generators in the third section anti-commute with the measurement operators $\{X_j\}_{j\in V_G}$ of the maximal extraction pattern $V_G = \{3, 5, \ldots, n-2\}$. Measuring these nodes and removing them results in the post-measurement state in **Tab.** 5.2.

After removing the X-support on the measured nodes of the odd-indexed generators (see sec. 2.3), they can be removed from the post-measurement state. This results in the $|V_G|$-qubit post-measurement state with generators listed in **Tab.** 5.2. These generators closely resemble those of the GHZ state as defined in Def. 17, but are not exactly the same. A local Clifford operation, which in general depends on the measurement outcomes, maps the post-measurement state to the desired $|\mathrm{GHZ}_{V_G}\rangle$ state. The exact calculation of this local Clifford operation is presented in chapter B.

	1	2	4	6	$\cdots$	$n-3$	$n-1$	n	ϕ
g_1	X	Z			...				$+1$
g_3		Z	Z		...				m_3
g_5			Z	Z	...				m_5
...					...				...
g_{n-2}					...	Z	Z		m_{n-2}
g_n					...		Z	X	$+1$
g_2'	Z	X	X		...	X	X	Z	$+1$

Table 5.2: After performing all measurements and removing the measured nodes, only those generators from **Tab.** 5.1 that commute with the measurement operators remain, which now carry the measurement outcomes $\{m_j = \pm 1\}$ as a phase ϕ. The post-measurement state is LC-equivalent to the target GHZ state (see Def. 17); the exact local Clifford operation is detailed in chapter B.

5.4 | Reduction of other patterns

Section 5.2 showed that many extraction patterns are not possible due to Thm. 1, and sec. 5.3 showed that extraction is indeed possible for a specific pattern that is not prohibited by the same theorem, namely the maximal extraction pattern. As noted before, the only restriction on the extraction patterns is that of Thm. 1, so that all other patterns not prohibited by it are possible as well. This section details how any such other pattern can be interpreted as a reduction to the maximal pattern of a smaller linear cluster state. Section 5.4.1 details how any smaller linear cluster state can be extracted from $|L_n\rangle$, and sec. 5.4.2 subsequently details how any non-maximal extraction pattern can be seen, provided it is not prohibited by Thm. 1, as the maximal extraction pattern of such a smaller linear cluster state.

5.4.1 | Extracting smaller linear cluster states from $|L_n\rangle$

An important and useful feature of the linear cluster state is that $|L_{n-1}\rangle$ can be extracted from $|L_n\rangle$ by removing any choice of node. If the chosen node is the first or last, a Z-basis measurement obtains the desired result: such a measurement is represented by a deletion of the node, followed by a Z Pauli operation on node 2 or $n-1$ if the measurement outcome was $m = -1$ (see sec. 3.4.1).

Similarly, a Y-basis measurement on any internal node j results in a linear cluster state on the remaining nodes. A Y-basis measurement is rep-

resented by a local complementation on the node, followed by its deletion (see sec. 3.4.2), so the measurement Y_j first connects the nodes $j-1$ and $j+1$ by the local complementation, and subsequently removes node j. This means that the resulting graph is a line graph from $1, 2, \ldots, j-1, j+1, \ldots, n$, and the post-measurement state is LC-equivalent to the associated graph state.

To obtain the true desired $|L_{1,2,\ldots,j-1,j+1,\ldots,n}\rangle$ state, a correction using the operators $\sqrt{Z_{j-1}}^\dagger$ and $\sqrt{Z_{j+1}}^\dagger$ on the two neighbours of the measured nodes is needed. Moreover, a subsequent Z operator to these two nodes has to be applied if the measurement outcome was $m_j = -1$ (see sec. 3.4.2).

These internal and external measurements and corrections can be repeated to extract a linear cluster state on any subset of the original nodes. As an example, consider the linear cluster state $|L_{1,4,5,6,7}\rangle$ that is to be extracted from the state $|L_{1,2,3,4,5,6,7}\rangle$. One could first extract the state $|L_{1,3,4,5,6,7}\rangle$ by performing a Y-basis measurement of node 2 and applying the associated correction operators. Subsequently, the state $|L_{1,4,5,6,7}\rangle$ can be extracted by performing a Y-basis measurement of node 3, including the suitable correction operators.

However, note that the correction operator $\sqrt{Z_3}^\dagger$ associated with the first measurement essentially rotates the subsequent measurement of node 3 towards the X basis (although the original correction operators still apply for the second measurement). If a set $\{j+1, \ldots, j+k\}$ of consecutive (internal) nodes is to be removed, these rotations give an alternating Y-X-Y-$\ldots$ pattern for the measurement bases. The correction operators are then $(\sqrt{Z_j}^\dagger)^k$ and $(\sqrt{Z_{j+k+1}}^\dagger)^k$, i.e. a $\sqrt{Z}^\dagger$ operator on the nodes j and $j+k+1$ for every node that has been measured. The measurement outcomes still need to be accounted for by potentially applying another Z Pauli operation to the same two nodes. The calculation to determine if these need to be applied is tedious but straightforward: it reveals that the correction needs to be applied only if the collection of measurement outcomes has odd parity. Python code that calculates the correction operators for any selection of nodes and measurement outcomes can be found in Sup. **[sB]** ([127]).

5.4.2 | Reduction of non-maximal extraction patterns

Turning back to extracting GHZ states, the above discussion can help with any extraction pattern that is not directly prohibited by Thm. 1. Any such pattern can be seen as the maximal extraction pattern for a smaller linear cluster state. This smaller linear cluster state can be obtained from the original n-partite resource graph state as described in sec. 5.4.1.

Fig. 5.3 shows such a non-maximal extraction pattern, and how it relates to a smaller linear cluster state. However, the extraction pattern V_G has only one 2-island, so it cannot be a maximal extraction pattern exactly. How this can be addressed is discussed below.

In the (general) case that an extraction pattern has no left 2-island (i.e. its

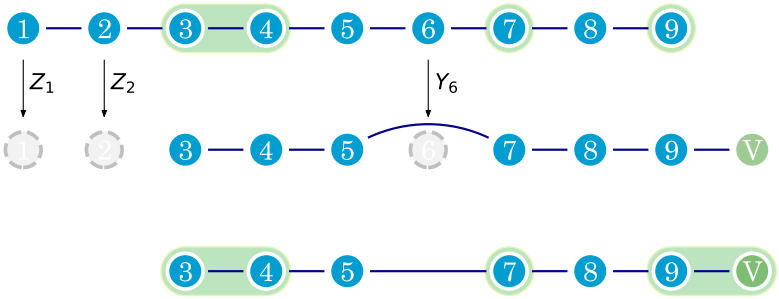


Figure 5.3: Any extraction pattern that is not impossible by Thm. 1 is possible, by viewing it as the maximal extraction pattern of a smaller 'virtual' linear cluster. The highlighted extraction pattern $V_G = \{3, 4, 7, 9\}$ is not the maximal extraction pattern for the state $|L_9\rangle$, but it is for the state $|L_{3,4,5,7,8,9,V}\rangle$, where node V is an additional 'virtual' node. This smaller linear cluster state can be obtained from $|L_9\rangle$ by two Z-basis and one Y-basis measurement (see sec. 5.4.1).

leftmost node, l, has no direct neighbour), it can not be seen as a maximal extraction pattern for any smaller linear cluster state. However, a larger GHZ state can then first be realized, after which a single-qubit measurement realizes the desired target GHZ state on V_G. Indeed, including the node $l-1$, the left neighbour of node l, results in an extraction pattern $V_G' = \{l-1\} \cup V_G$ that can be seen as the maximal extraction pattern for a (potentially smaller) linear cluster state. This extraction pattern gives the state $\left|\mathrm{GHZ}_{V_G'}\right\rangle$. A subsequent X-basis measurement on node $l-1$ removes this node from the larger GHZ state, so that the desired target state $|\mathrm{GHZ}_{V_G}\rangle$ is obtained. If the outcome of this measurement is $m_{\{l-1\}} = -1$, a Z_l correction is needed.

When the leftmost node is $l = 1$, it does not have a left neighbour. However, since it will be measured anyway, one can introduce a 'virtual' node 0, which would be measured in the X-basis if it existed. The case for when V_G does not have a right 2-island follows similarly.

These considerations allow to analyse the pattern in **Fig.** 5.3: the resource state is the 9-qubit linear cluster state $|L_9\rangle$, and the desired extraction pattern is $V_G = \{3, 4, 7, 9\}$. V_G is not the maximal extraction pattern for $|L_9\rangle$, but it is for the state $|L_{3,4,5,7,8,9,V}\rangle$, where node V is the 'virtual' node that has been virtually introduced. This state can first be realized from $|L_9\rangle$ by Z-basis measurements of nodes 1 and 2 and a Y-basis measurement of node 6, followed by the associated corrections. $|L_{3,4,5,7,8,9,V}\rangle$ can then be measured according to the maximal extraction pattern, obtaining the desired GHZ state.

Any pattern that is not prohibited by Thm. 1 can be seen as a maximal pattern for a smaller linear cluster state, so that it is possible, which completely characterizes all target GHZ states that can be extracted from a

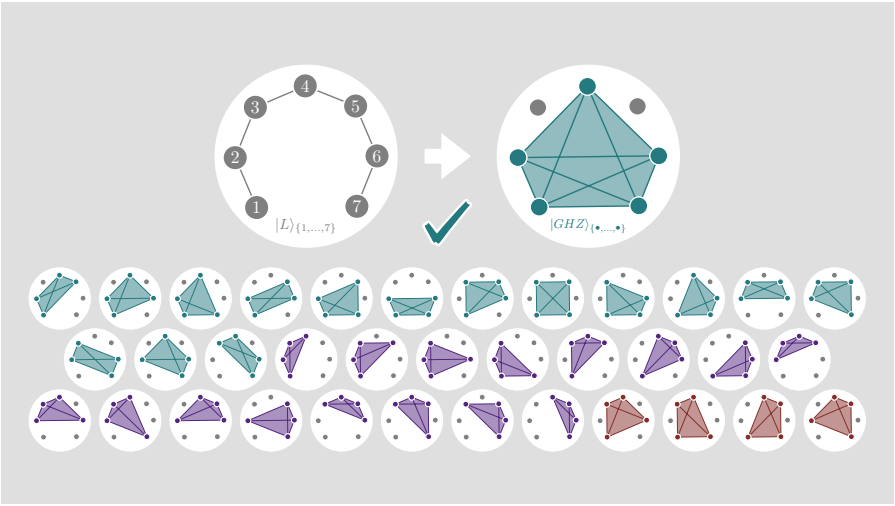


Figure 5.4: **(Top row, left)** The linear cluster state $|L_7\rangle$ is the resource graph state, the desired target graph state is the GHZ state. **(Top row, right)** The maximal extraction pattern obtains the largest GHZ state, containing five qubits. **(Bottom rows)** All extraction patterns V_G with four nodes. Those that are possible have been highlighted in green, those that are impossible through Cor. 1 have been highlighted in violet, and those that are impossible directly through Thm. 1 are highlighted in red.

resource linear cluster state. **Fig.** 5.4 contains the 7-qubit linear cluster state as the resource graph state, the maximal extraction pattern for this resource state, and all the choices of extraction patterns V_G of size four. The possible and impossible extraction patterns are highlighted in different colours.

5.5 | Implementations

To complement the theoretical analysis and to experimentally demonstrate the GHZ state extraction, implementations on `IBMQ` hardware were performed using the `qiskit` SDK [131]. More specifically, linear cluster states of size $n \in \{5, 7, \ldots, 19\}$ were prepared on both the `IBMQ Cairo` [132] and `IBMQ Mumbai` [133] devices, from which GHZ states of size $n = \{4, 5, \ldots, 11\}$ were extracted through the maximal extraction pattern.

The traditional method to prepare linear cluster states, using qubits prepared in the $|+\rangle$ state on which the gates $C_Z^{(i,i+1)}$ are applied, is not suitable for the native gate set of the used devices. Therefore, the preparation circuit is compiled towards the gate set of the devices as shown in **Fig.** 5.5 for three qubits; the generalisation for higher (odd) n is straightforward.

The last layer in the circuit, consisting of Hadamard operations on the

odd-numbered qubits, is not actually implemented. The maximal extraction pattern dictates that the nodes not in V_G (i.e. $\{3, 5, \ldots, n-2\}$) are measured in the X-basis, which on the `IBMQ devices` is implemented as another Hadamard operation followed by a Z-basis measurement. These Hadamard operations would cancel out against those in the last layer of the right circuit in **Fig.** 5.5, so they are both omitted. Similarly, the analysis in chapter B shows that the local Clifford corrections to obtain the GHZ state involve a Hadamard operation on the first and last qubit; these two operations are cancelled out by those in the last layer of the right circuit in **Fig.** 5.5, which means that for the first and last qubit this layer can be omitted as well.

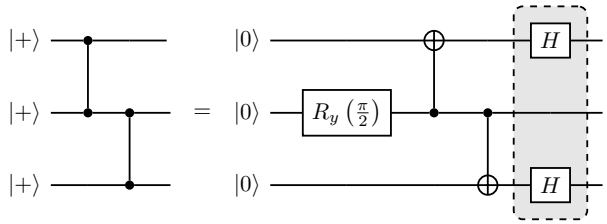


Figure 5.5: The left circuit is the traditional circuit that prepares the linear cluster state $|L_3\rangle$. However, the `IBMQ Montreal` device does not allow for native CZ or Hadamard gates, but only for CX gates and rotations along the Pauli axes. The right circuit is a hand-compiled circuit that prepares the $|L_3\rangle$ state with only native gates. The last layer of the compiled circuit, containing Hadamard operations, is not implemented but emulates the X-basis measurements from the extraction pattern and the associated corrections. The generalized circuit for larger (odd) n is straightforward.

To test the performance of the extraction, the fidelity of both the prepared linear cluster states and extracted GHZ states is estimated. More specifically, a lower bound on the fidelity is estimated by performing selected measurements of the stabilizer elements. For the linear cluster state, this estimate is provided by an analysis similar to [134], originally inspired by [135]. More specifically, following (2.10) the fidelity of the prepared state ρ with the linear cluster state (up to the last layer of Hadamard rotations) can be calculated as:

$$F(\rho, |L_n\rangle) = \mathrm{tr}\left[\rho \prod_j \frac{\mathbb{I} + g_j}{2}\right], \tag{5.3}$$

where the g_j are the generators of the rotated linear cluster state, i.e. without the last layer of **Fig.** 5.5. They can be grouped into a set of 'odd' generators $G_o^L = \{Z_{i-1}Z_iZ_{i+1}\}_{\text{odd } i}$ and 'even' generators $G_e^L = \{X_{i-1}X_iX_{i+1}\}_{\text{even } i}$, where it is to be understood that $\sigma_z^0 = \sigma_z^{n+1} = 1$. These two sets generate two different subgroups of the stabilizer, referred to as the 'odd' and 'even'

subgroups:

$$\begin{aligned} \mathcal{S}_o &= \langle G_o^L \rangle \subset \mathcal{S}, \\ \mathcal{S}_e &= \langle G_e^L \rangle \subset \mathcal{S}. \end{aligned} \tag{5.4}$$

Note that the elements of $\mathcal{S}_o$ and $\mathcal{S}_e$ only consist of Z and X operators, respectively, which is very useful in the estimation method.

By writing $G_o = \prod_{g \in G_o^L} \frac{\mathbb{I}+g}{2}$, and similarly for the even generators, (5.3) can be written as:

$$\begin{aligned} F(\rho, |L_n\rangle) &= \operatorname{tr}\left[G_o G_e \rho\right] \\ &= \operatorname{tr}\left[G_o \rho\right] + \operatorname{tr}\left[G_e \rho\right] - \operatorname{tr}\left[\mathbb{I}\rho\right] + \operatorname{tr}\left[K\rho\right], \end{aligned} \tag{5.5}$$

where $K = (\mathbb{I} - G_o)(\mathbb{I} - G_e)$. The term $\operatorname{tr}\left[G_o\rho\right]$ can be written in terms of the elements of $\mathcal{S}_o$:

$$\operatorname{tr}\left[G_o\rho\right] = \mathbb{E}\left[G_o\right] = \frac{1}{2^{|\mathcal{S}_o|}} \sum_{\sigma \in \mathcal{S}_o} \operatorname{tr}\left[\rho\sigma\right], \tag{5.6}$$

and $\operatorname{tr}\left[G_e\rho\right]$ follows similarly. Moreover, K is positive semidefinite so that the last term in (5.5) can be discarded. This results in a lower bound for the fidelity (using $\operatorname{tr}\left[\mathbb{I}\rho\right] = 1$):

$$F(\rho, |L_n\rangle) \geqslant \mathbb{E}\left[G_o\right] + \mathbb{E}\left[G_e\right] - 1. \tag{5.7}$$

Rewriting the fidelity like this is very helpful, because it is relatively straightforward to obtain (estimates for) $\mathbb{E}\left[G_o\right]$ and $\mathbb{E}\left[G_e\right]$. Still, either subgroup $\mathcal{S}_{o(e)}$ contains an exponential number of elements, so measuring the terms $\operatorname{tr}\left[\rho\sigma\right]$ one-by-one is undesirable. Moreover, the elements of $\mathcal{S}_o$ and $\mathcal{S}_e$ are multi-qubit operators, whose measurements generally involve multi-qubit operations. However, these measurements can be reconstructed from single-qubit measurements, which is explained in chapter C.

As noted before, the odd subgroup $\mathcal{S}_o$ consists of only Z (and $\mathbb{I}$) operators. The measurements of all the elements of $\mathcal{S}_o$ can therefore be reconstructed from simultaneous measurements of every single qubit in the Z-basis. Similarly, every measurement associated with the even subgroup can be reconstructed from the measurement setting where every single qubit is measured in the X-basis. Using the analysis in chapter C, it follows that with just two measurement settings the fidelity can be estimated; for both settings the measurements have to be repeated so that the terms $\mathbb{E}\left[G_o\right]$ and $\mathbb{E}\left[G_e\right]$ can be estimated.

Similar to the linear cluster state, the generators of the GHZ state can be grouped into $G_o^G = \{X_{V_G}\}$ and $G_e^G = \{Z_j Z_{j+1}\}_{j \in V_G}$ (where the node $j+1$ indicates the right neighbour of j within the extraction pattern). Note that the odd group G_o^G consists of a single generator. Combining the associated measurements with the measurements necessary for extraction, the fidelity

of the GHZ state can be estimated with just two measurement settings as well. After the measurements for the extraction of the GHZ state, post-measurement correction operators need to be applied as detailed in chapter B. This correction consists of applications of X operators to some selection of the nodes in V_G; this selection depends on the outcomes of the measurements during the extraction.

However, immediately after applying these X operators the qubits are measured in the X- or Z-basis. For the X-basis measurements, the X correction operator has no effect. For the Z-basis measurements, the only effect that these X correction operators have on the measurement outcomes is that these are flipped. It follows that instead of actually applying the X correction operator, the qubits can be measured in the Z-basis without them, after which the $+1$ and -1 outcomes are exchanged; this technique generalizes to the *Pauli frame* [76], and is closely related to a similar technique discussed in sec. 9.5. Thus, the X-basis measurements of the qubits *not* in V_G for the extraction, and the X- or Z-basis measurements of the qubits in V_G for the fidelity estimation, can be performed simultaneously.

All measurement settings were repeated a total of 32000 times to gather enough statistics to obtain a good estimate of the terms in (5.7); the results are presented in **Fig.** 5.6.

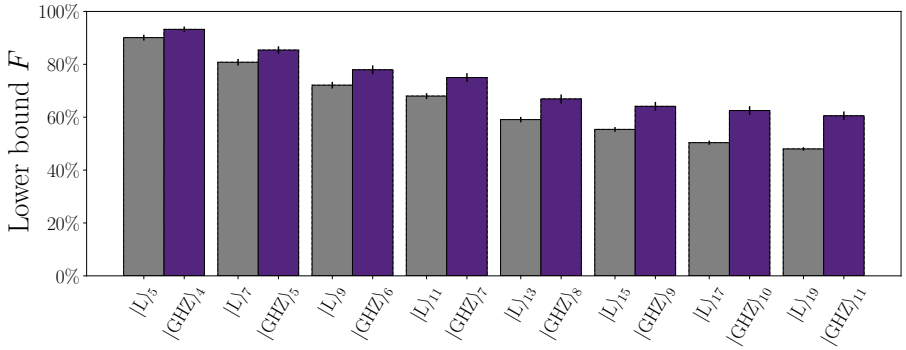


Figure 5.6: A lower bound for the fidelity obtained for both the resource graph state $|L_n\rangle$ and the target graph state GHZ. Linear cluster states of size $n = \{3, 5, \ldots, 19\}$ are prepared using the circuit from **Fig.** 5.5, from which GHZ states of size $|V_G| = \{4, 5, \ldots, 11\}$ are extracted through the maximal extraction pattern. Due to a technical detail in the estimation method of the fidelity, the bound is less strict for the linear cluster state than for the GHZ state. This results in an estimated fidelity for the linear cluster state that is lower than that of the associated GHZ state.

A caveat with the described estimation method is that for the GHZ state the odd generators give a subgroup $\mathcal{S}_o = \langle X_{V_G} \rangle = \{\mathbb{I}, X_{V_G}\}$, with just two elements. The 'even' subgroup, and both subgroups for the linear cluster state,

are all considerably larger and growing with n. Due to the small size of the odd subgroup of the GHZ state, the term $\mathrm{tr}\,[\mathbb{I}\rho] = 1$ has a relatively large impact in (5.6). Noise and imperfections affect all terms of the sum where the stabilizer element is not $\mathbb{I}$, so that if the subgroup is larger, it is (relatively) more affected by noise. Especially for larger n this effect becomes more pronounced, so that the estimates of the GHZ nodes become considerably better than the estimates of the linear cluster states, even though the former is extracted from the latter. This is clearly visible in **Fig.** 5.6.

One option to solve this issue is by taking the sum in (5.6) over $\mathcal{S}_o \setminus \{\mathbb{I}\}$ and $\mathcal{S}_e \setminus \{\mathbb{I}\}$. By removing the identity elements, all elements in the sum are equally affected by noise. This results in (estimates of) a lower bound that are generally less strict (i.e. *worse*), but give fairer results between the linear cluster states and the GHZ states. **Fig.** 5.7 contains the estimates with this adapted method, which indeed show more equal, but generally worse fidelities than **Fig.** 5.6.

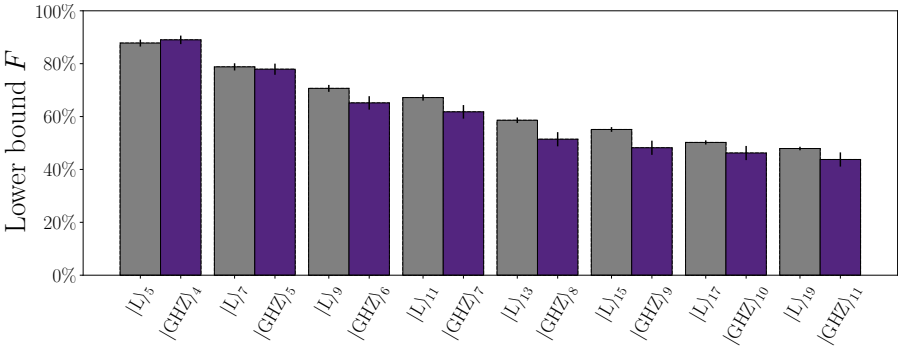


Figure 5.7: The bounds in this figure use an adapted method of estimating the fidelities, so that the bounds give a more equal estimate between the resource and target graph states. However, at the same time it gives a lower, less strict bound than that of **Fig.** 5.6.

5.6 | Conclusion

This chapter has shown that, for specific choices of resource and target states, it can be decided if extraction is possible. It has shown an upper bound to the size of any GHZ state that can be extracted from linear cluster states, and has completely characterized what GHZ states can indeed be obtained. Generalizing to other specific target or resource states is not straightforward by using the methods presented in this chapter. However, as was already pointed out in the discussion of Pub. **[F]**, the methods do generalise easily to *ring graph states*, i.e. the linear cluster state with an additional edge between

nodes 1 and n. In such a case only a single 2-island is possible, so that the upper bound becomes $|V_G| = \lfloor {}^{n+1}/_{2} \rfloor$.

Although in this chapter the linear cluster state was explicitly chosen as the resource state, any state that is locally equivalent to it is automatically suitable for GHZ state extraction as well. Indeed, that state can first be rotated to the linear cluster state by a local operation, after which the extraction can be performed as explained in this chapter.

Technically, this chapter has considered only local Clifford operations and single-qubit Pauli measurements, instead of the more general case of local unitary operations and single-qubit measurements in other bases. It is straightforward to show that the LU-orbit and LC-orbit of the GHZ state coincide, as they do for the linear cluster state (see sec. 4.4.2). It follows that no generality is lost by considering only local Clifford operations instead of the more general local unitary operations. Although not proven yet, it seems as if a similar argument can be made for the single-qubit Pauli measurements, so that no generality is lost by not allowing for single-qubit measurements in *any* basis.

Because any state in the LU-orbit $\mathcal{O}^{\mathrm{LU}}(|L_n\rangle)$ is suitable for GHZ state extraction, it is very useful to determine if a given graph state $|G\rangle$ is part of this LU-orbit. More generally, characterising different states by their associated LU-orbits is helpful to determine many useful properties. The methods of chapter 6 work towards this goal.

6

Characterizing Entanglement

> The distinguishing mark of the orbit is the marginal, the instrument with which it does all its mischief.
>
> George Orwell, *Animal Farm* (paraphrased)

The previous chapter showed that it is possible to characterize extraction patterns, but only for highly specific cases like the extraction of GHZ states from linear cluster states. It was assumed that the resource state was exactly the linear cluster state, somehwat restricting the usability of the results. Nevertheless, there are many other states that can be used as resource states for GHZ extraction; at least any state that is LU-equivalent to the linear cluster state is guaranteed to be suitable. Indeed, first the state can be rotated to the linear cluster state by a local unitary operation, after which the GHZ extraction can be performed; of course, states from other orbits could be suitable for the extraction as well.

Following the discussion in sec. 4.2, any stabilizer state is LC-equivalent to a graph state, so this chapter focusses solely on the equivalence of graph states. In general, as discussed in chapter 4, graph states from the same orbits

can be seen as containing the same type of entanglement, and thus (roughly speaking) as the same resource in networking tasks. It is therefore incredibly helpful to be able to identify if a set of states belongs to the same LU-orbit, or even characterize states by what specific LU-orbits they belong to.

This chapter presents the contents of Pub. **[G]** ([55]), that presented methods to study the LU-equivalence of graph states, and introduced methods to characterize LU-orbits. This characterization can be computed from any representative of any LU-orbit, resulting in a type of 'identifier' that is constant for every element of the LU-orbit the representative belongs to. Elements of the same LU-orbit are then evaluated to have the same identifier, meaning that the identifier is *invariant* for all the elements of an LU-orbit. Hence, this identifier needs to be derived from an *LU-invariant*, i.e. a property of any stabilizer state that does not change when the state undergoes a local unitary transformation.

In a slightly more abstract, theoretical setting, one may wish to determine if states are from the same entanglement class (see sec. 4.3) instead of LU-orbit, where stabilizer states that are equivalent under permutations would additionally evaluate as having the same identifier. Thus, an identifier that is not only invariant under LU operations, but additionally invariant under permutations needs to be obtained. Beyond the equivalence of LU-orbits, Pub. **[G]** addressed the equivalence of graph states in this setting as well, and the characterisation of entanglement classes.

These identifiers are designed in such a way that elements of an LU-orbit or entanglement class are always evaluated to have the same identifier, essentially labelling the LU-orbit or entanglement class with that identifier. Ideally, these identifiers are unique, in the sense that two *different* LU-orbits or entanglement classes are always labelled with different identifiers, so that they can be distinguished. This is not always the case, so the performance of the identifiers must be assessed.

For both LU-orbits and entanglement classes, the identifiers that are introduced in this chapter are derived from a single specific LU-invariant: the dimension of the reduced stabilizer (see Def. 12). How to compute the dimension for a graph state is discussed in sec. 6.1. Section 6.2 shows that the stabilizer dimension can indeed be used as an LU-invariant. In sec. 6.3 various identifiers are derived from this invariant, for both LU-orbits and entanglement classes. The performance of the identifiers is assessed in sec. 6.4 by two different performance metrics. This assessment makes use of an online database of (representatives of) LC-classes (see sec. 4.3) found in [136], the supplementary material of [115].

It can happen that two (representatives of) different LC-orbits are calculated to have identical values for all introduced identifiers. Interestingly, this does not necessarily mean that the identifiers have failed in faithfully determining LU-equivalence. Indeed, although the representatives are from different *LC-orbits*, they can in principle still be LU-equivalent, so that their

identifiers are equivalent as well. This would render them counterexamples to the LU-LC-conjecture (see sec. 4.4.2). Examples of two different LC-orbits with the exact same identifiers can indeed be found; the smallest example consists of two graphs of nine nodes, presented in sec. 6.5. However, they are indeed LU-inequivalent as well. Because the identifiers fail to tell them apart, another method is necessary to show this, which is discussed in the same section. Section 6.6 discusses the efficiency and scaling properties of calculating the introduced identifiers. Finally, sec. 6.7 concludes the chapter.

Pub. **[G]** ([55]) introduced and discussed more methods regarding its topics than are addressed in this chapter. The interested reader is referred to the publication. Some of these results were presented in [137] as well.

6.1 The reduced stabilizer for a graph state

Eq. (2.25) showed that the reduced state of a stabilizer state and its associated reduced stabilizer are closely related. Hence, to calculate the reduced state of a graph state, one could first calculate its reduced stabilizer. Calculating this could in principle be done by first listing every element in the original stabilizer, and selecting only those with the correct support. However, this is a tedious process as the stabilizer grows exponentially in size.

This section introduces another method to calculate the reduced stabilizer for graph states, which is a much more efficient approach. More specifically, let $|G\rangle$ be an arbitrary graph state on the qubits $V = [n]$ with stabilizer $\mathcal{S}$ and generators $\{g_i = X_i Z_{\mathcal{N}_i}\}_{i\in V}$. The goal is to find the reduced stabilizer $\mathcal{S}_M$ for an arbitrary choice of $M \subset V$, with $k = |M|$ the size of M. For this, it needs to be determined, for every $P \in \mathcal{S}$, if $\mathrm{supp}(P) \subseteq M$ (see Def. 12).

To this effect, (2.4) can be used to represent P by a sequence of bits $\{b_1, b_2, \ldots, b_n\}$, where b_i 'encodes' if the generator g_i is 'used' or not. Let $B = \{i \in V | b_i = 1\}$ represent the set of generators that are 'used'. Then, the operator P can be decomposed into an X- and Z-part separately, in terms of B:

$$P = \prod_{i\in B} X_i Z_{\mathcal{N}_i} \sim \Big(\prod_{i\in B} X_i\Big)\Big(\prod_{i\in B} Z_{\mathcal{N}_i}\Big) = X_B\Big(\prod_{i\in B} Z_{\mathcal{N}_i}\Big), \tag{6.1}$$

where X_B represents an X operator on every node in B and $\sim$ indicates that equality holds up to a phase ± 1. Ultimately this representation is used to determine if the support of P is contained in M, for which this phase is irrelevant.

From (6.1) it follows directly that for any $P \in \mathcal{S}$ that is in the reduced stabilizer, it must hold that $B \subseteq M$. Indeed, otherwise P would have support (at least with X) on every node $B \setminus M$, which are nodes in $M^\perp$.

If the neighbourhood of every generator that is 'used' (i.e. those in B) is additionally contained in M, then it is straightforward to see that $\mathrm{supp}(P) \subset M$. Consider the marginal $M_1 = \{1, 2\}$ from **Fig.** 6.1, highlighted in green.

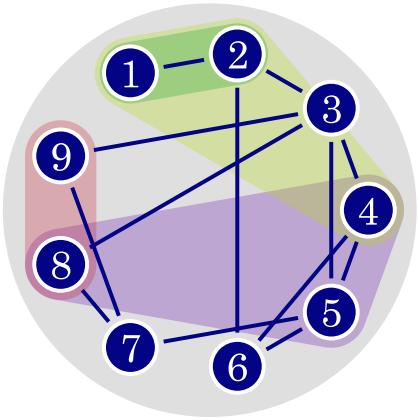


Figure 6.1: A graph with a selection of marginals highlighted; all these marginals are non-trivial, so that $d_M \geqslant 1$. The dimensions of the four highlighted marginals $M_1 = \{1, 2\}$, $M_2 = \{8, 9\}$, $M_3 = \{4, 5, 8\}$ and $M_4 = \{1, 2, 4\}$ are all analysed in the main text.

Node 1 has $\mathcal{N}_1 = \{2\}$, so taking $B = \{1\}$ results in the Pauli operator $P = X_1 Z_2$, whose support is contained in M. Thus, the operator $X_1 Z_2$ (as an element of $\mathcal{P}_2$) is part of the reduced stabilizer $\mathcal{S}_{M_1}$.

However, generators that have Z support outside of M do not necessarily lead to stabilizer elements that cannot be elements of $\mathcal{S}_M$. Indeed, the Z support from different generators can cancel out through the identity $Z^2 = \mathbb{I}$. Consider the marginal $M_2 = \{8, 9\}$ from **Fig.** 6.1, highlighted in red. Neither the generator $g_8 = X_8 Z_3 Z_7$ nor the generator $g_9 = X_9 Z_3 Z_7$ has its support contained in M_2. However, their product $g_8 g_9 = X_8 X_9$ has no Z operators in $M_2^\perp$, because every Z operator outside of M_2 appears an even number of times, and has thus cancelled out. It follows that the stabilizer element $g_8 g_9$ has its support contained in M_2, and is therefore (as an operator from $\mathcal{P}_2$) part of the reduced stabilizer $\mathcal{S}_{M_2}$.

General case

The case for general M follows readily. Let $P \in \mathcal{S}$ be any stabilizer element, and let B be the selection of generators (i.e. nodes) that represent P through (6.1). It is the goal to determine if $P \in \mathcal{S}_M$ or not. Any time a node $j \in M^\perp$ is contained an even number of times in the neighbourhoods $\mathcal{N}_i$ of the nodes in B, the Z operators on node j cancel out. Thus, for the selection B, Z operators only remain on those nodes that are included an odd number of times in the neighbourhoods of the nodes in B.

This set of nodes is exactly the symmetric difference of all the neighbourhoods combined, denoted (with abuse of notation) ΔB. Using this insight, (6.1) can be rewritten as:

$$P \sim X_B Z_{\Delta B}. \tag{6.2}$$

It can be concluded that the elements of $\mathcal{S}_M$ are exactly those that can be written as (6.2) with both $B \subseteq M$ (for the X-part of the support), and with $\Delta B \subseteq M$ (for the Z-part of the support).

At this point it is extremely useful to represent these neighbourhoods as the columns η_i of the adjacency matrix Γ of the graph G (see (3.6)). The symmetric difference ΔB of all neighbourhoods combined then is represented by the binary vector η_B:

$$\eta_B = \bigoplus_{i \in B} \eta_i, \tag{6.3}$$

where the addition is performed over the binary field; now $\eta_B(j) = 1$ only for those $j \in \Delta B$. Thus, $\mathrm{supp}(P) \subset M$ if and only if $B \subseteq M$ (for the X-part), and $\eta_B(j) = 0 \quad \forall j \in M^\perp$ (for the Z-part).

W.l.o.g. M can be taken to be the first k nodes of V. The vector η_B can then be split into its parts regarding M and $M^\perp$. The necessary condition then becomes that the latter part is equal to the zeros vector:

$$\eta_B = \left[\begin{array}{c} \eta_B^{(M)} \\ \hline \eta_B^{(M^\perp)} \end{array}\right] = \left[\begin{array}{c} \eta_B^{(M)} \\ \hline \mathbf{0} \end{array}\right], \tag{6.4}$$

where $\mathbf{0}$ indicates a zeros vector of adequate length. Note that the equality $\eta_B^{(M^\perp)} = \mathbf{0}$ does not hold in general for any B, but only whenever the Z-support is indeed contained in M.

Similarly, the adjacency matrix Γ itself can be written in block form:

$$\Gamma = \left[\begin{array}{c|c} \Gamma^{(M,M)} & \Gamma^{(M,M^\perp)} \\ \hline \Gamma^{(M^\perp,M)} & \Gamma^{(M^\perp,M^\perp)} \end{array}\right], \tag{6.5}$$

where the matrix $\Gamma^{(A,B)}$ indicates the sub-matrix of Γ with the rows and columns indexed by A and B, respectively. The sub-matrix $\Gamma^{(M^\perp,M)}$ is exactly the matrix whose columns are the vectors $\eta_i^{(M^\perp)}$ (i.e. η_i split into parts as in (6.4)):

$$\Gamma^{(M^\perp,M)} = \left[\eta_1^{(M^\perp)} \quad \eta_2^{(M^\perp)} \quad \dots \quad \eta_M^{(M^\perp)}.\right] \tag{6.6}$$

In other words, $\Gamma^{(M^\perp,M)}$ represents the $M^\perp$-part of the neighbourhoods η_i for every node in M.

Using (6.3), any $\eta_B^{(M^\perp)}$ can then be written as a linear combination of these columns:

$$\Gamma^{(M^\perp,M)} \mathbf{x}_B = \eta_B^{(M^\perp)}, \tag{6.7}$$

where $\mathbf{x}_B$ is the binary vector of length[1] k 'selecting' the nodes in B.

The Pauli P associated with B has support contained in M if $\eta_B^{(M^\perp)} = \mathbf{0}$ (see (6.4)). Combining this with (6.7), this results in the condition:

$$\Gamma^{(M^\perp,M)}\mathbf{x}_B = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{6.8}$$

It follows that every element $P \in \mathcal{S}$ corresponds to a unique vector $\mathbf{x}_B$ for which (6.8) holds. These vectors are exactly the elements of the kernel (over $\mathbb{F}_2^k$) of the $(n-k) \times (k)$ matrix $\Gamma^{(M^\perp,M)}$, so that determining the reduced stabilizer $\mathcal{S}_M$ is the same as finding the solutions to a set of $n-k$ linear equations.

This set of solutions forms a linear subspace of $\mathbb{F}_2^k$, and every element of it uniquely represents an element of the reduced stabilizer. By Def. 12 and (2.24), the dimension of this subspace, which is the nullity of $\Gamma^{(M^\perp,M)}$, is exactly the stabilizer dimension d_M. Since it is efficient to compute the nullity of a matrix, this provides an efficient method to determine the stabilizer dimension d_M of a graph state $|G\rangle$ for any marginal M.

Eq. (6.8) and its insights can, beyond calculating d_M, additionally be used to obtain representations of the actual elements of the reduced stabilizer $\mathcal{S}_M$. More specifically, a basis $\{b_l\}_{l=1}^{d_M}$ of the above subspace exactly represents a set of generators $\{g_l^{(M)}\}$ for $\mathcal{S}_M$:

$$g_l^{(M)} = \prod_{i \in M | b_l(i)=1} g_i, \tag{6.9}$$

where $b_l(i)$ is the i-th element of the vector b_l.

Beyond the two examples already discussed earlier, **Fig. 6.1** presents two other examples of non-trivial marginals. The marginal $M_3 = \{4,5,8\}$, highlighted in blue, has no generators whose support is contained within M_3. Similarly, any product of two generators results in a Pauli operator that has support outside of M_3 as well. However, the product $g_4 g_5 g_8 = X_4 X_5 X_8$ results in a Pauli operator with support contained in M_3, which means that the marginal is not trivial. There is only one surviving element, so that $d_{M_3} = 1$.

The marginal $M_4 = \{1,2,4\}$, highlighted in yellow, has multiple non-trivial elements, however. The generator $g_1 = X_1 Z_2$ remains, and so does the product $g_2 g_4 = Z_1 X_2 X_4$. These two elements can be seen as a generating set for $\mathcal{S}_{M_4} = \{\mathbb{I}, g_1, g_2 g_4, g_1 g_2 g_4\}$, so that the marginal dimension for M_4 is $d_{M_4} = 2$.

[1] The length of $\mathbf{x_B}$ is k because it has already been (implicitly) assumed that $B \subset M$.

6.2 The rank of reduced states as an invariant

It is straightforward to show that the rank of any marginal cannot change under local unitary operations, as the following theorem shows (also shown in [3] and Pub. **[G]** ([55])).

Theorem 2. *Let ρ and σ be two LU-equivalent stabilizer states with respective stabilizers $\mathcal{S}^\rho$ and $\mathcal{S}^\sigma$, so that $\rho = U\sigma U^\dagger$ for some $U \in \mathcal{L}^{\mathcal{U}}$. Then for any choice of subset $M \subset \{1, 2 \ldots, n\}$, their reduced states have equal rank:*

$$\mathrm{rnk}(\rho_M) = \mathrm{rnk}(\sigma_M). \tag{6.10}$$

This rank is called the marginal rank *(w.r.t. M).*

Furthermore, the reduced stabilizers $\mathcal{S}^\rho_M$ and $\mathcal{S}^\sigma_M$ have equal dimension as well:

$$d^\rho_M = d^\sigma_M. \tag{6.11}$$

Proof. Because $U \in \mathcal{L}^{\mathcal{U}}$, it holds that $U = U_M \otimes U_{M^\perp}$ for any choice of M. ρ_M can easily be computed in terms of the reduced state σ_M:

$$\rho_M = \mathrm{tr}_{M^\perp}\left[\rho\right] = \mathrm{tr}_{M^\perp}\left[\left(U_M \otimes U_{M^\perp}\right)\sigma\left(U_M \otimes U_{M^\perp}\right)^\dagger\right] = U_M \sigma_M U_M^\dagger. \tag{6.12}$$

Thus, the reduced states ρ_M and σ_M are related by a unitary operation. It follows directly that their rank is identical. As M was chosen arbitrarily, it holds for any reduced state of ρ and σ. From (2.27) it follows that the dimensions of the reduced stabilizers are equal as well. □

Thm. 2 shows that the rank of the reduced state is indeed an LU-invariant for any choice of M. It follows that having the same marginal rank or dimension for *every* choice of M is a necessary condition for any pair of states to be LU-equivalent. Together with the analysis of sec. 6.1, this allows for an easy to compute method to determine if graph states are LU-inequivalent.

One example is given in **Fig.** 6.2. The three graphs from the figure are all from different LU-orbits, because their highlighted marginals have different dimensions.

6.3 Identifiers derived from the rank invariant

Fig. 6.2 shows that it can be very useful to consider more than one marginal at the same time to decide on LU-equivalence, especially when more than two graph states are to be considered. In general it is useful to categorize the marginal dimensions of a graph in a consistent approach, to facilitate comparisons of graphs regarding LU-equivalence. A first concept that offers such categorization is the *marginal list*, which can be defined for any marginal size k.

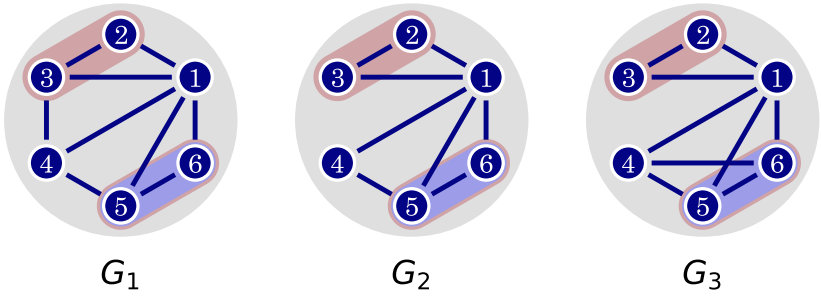


Figure 6.2: Three 6-qubit graph states $|G_1\rangle, |G_2\rangle$ and $|G_3\rangle$, where two different marginals are highlighted. The three graph states all belong to different LU-orbits. This follows from an inspection of the highlighted marginals. $|G_1\rangle$ has for both highlighted marginals $d_M = 0$. $|G_2\rangle$ has that the red marginal is non-trivial with $d_M = 1$, but the blue marginal is trivial. For $|G_3\rangle$, both highlighted marginals are non-trivial with $d_M = 1$.

Definition 26. *For an n-node graph G with vertex set V, and marginal size $k < n$, the k-body* marginal list l_k^G *is the length-$(k+1)$ vector*

$$l_k^G = \begin{bmatrix} L_0^{k,G} & L_1^{k,G} & \dots & L_k^{k,G} \end{bmatrix}, \tag{6.13}$$

where $L_i^{k,G}$ (for $0 \leqslant i \leqslant k$) is the number of all k-body marginals $M \subset V$ with stabilizer dimension $d_M = i$:

$$L_i^{k,G} = |\{M \subset V | |M| = k,\, d_M = i\}|. \tag{6.14}$$

When context permits G is dropped, so that the marginal list is written l_k.

In other words, the marginal list l_k^G is a vector of length $k+1$, where the i-th entry is the number of k-body marginals of G with marginal dimension i. Using Thm. 2 it is straightforward to see that the marginal lists of two graph states from the same LU-orbit are identical, which means that the marginal list l_k^G can function as an identifier of LU-orbits. Note that for connected graphs the last entry in the vector always equals zero. Furthermore, note that the sum of the entries of l_k^G is always equal to $\binom{n}{k}$, the total number of k-body marginals.

The marginal list l_k^G can distinguish many different LU-orbits (e.g. l_2^G is different for all three graphs from **Fig.** 6.2), but cannot represent *where* the marginals of a given dimension are. This means that it can fail to distinguish graph states even though they are LU-inequivalent. Notably, two elements from a single entanglement class may be from different LU-orbits, but the marginal list will never be able to distinguish them. For example, **Fig.** 6.3 shows two graphs that are not in each others LU-orbit, even though their marginal lists l_k coincide for every k.

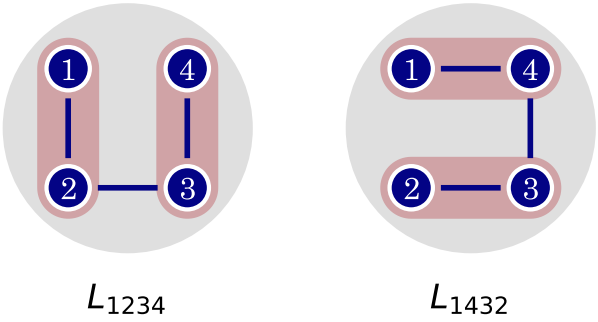


Figure 6.3: Two graphs that can not be distinguished by their two-body rank lists l_2^G, as they contain the same number of two-body marginals with $d_M = 1$. However, the positions of these marginals are different, so that their marginal tensors T_k^G are different. From this it can be concluded that the graphs are LU-inequivalent. Note that the two graphs are representatives of two different LU-orbits from the entanglement class of the $|L_4\rangle$ state, shown in **Fig.** 4.1.

It can therefore be necessary to not only categorize the number of marginals with every dimension, but additionally how they relate to one another, i.e. the 'positions' of the marginals w.r.t. the nodes. For this, it is useful to define the *marginal tensor*.

Definition 27. *For an n-node graph G with vertex set V, and a marginal size $k < n$, the (k-body)* marginal tensor T_k^G *is the tensor defined as*

$$T_k^G = \left(T_{i_1 \cdots i_k}\right)_{i_1,\ldots,i_k \in V}, \tag{6.15}$$

where the entries of the tensor are defined as

$$T_{i_1 \cdots i_k} := d_{\{i_1,\ldots,i_k\}}, \tag{6.16}$$

with double occurrences of nodes in the set $\{i_1, \ldots, i_k\}$ understood to be removed. When context permits G is dropped, so that the marginal tensor is written T_k.

In other words, the entry of the tensor indexed by $\{i_1, i_2, \ldots, i_k\}$ is exactly the dimension of the marginal of the graph state $|G\rangle$ given by $M = \{i_1, i_2, \ldots, i_k\}$. As noted in the definition, these indices $\{i_j\}$ are not necessarily unique, so that double occurrences are dropped. This leads to $1 \leqslant |M| \leqslant k$. In turn, this means that the marginal tensor of dimension k contains the dimensions of all marginals of size $\leqslant k$. Furthermore, note that the marginal tensor is supersymmetric.

Similarly to the marginal list, the marginal tensor functions as an identifier of LU-orbits: from Thm. 2 it is straightforward to see that the marginal tensors of two graph states from the same LU-orbit are identical. Because the positions of the marginals are represented by the tensor as well, it can distinguish strictly more graphs than the marginal lists. Indeed, the two graphs from **Fig.** 6.3, while having identical marginal lists, have different marginal tensors. From this it can be concluded that they are LU-inequivalent.

Distinguishing entanglement classes

The power of the marginal tensor T_k^G to additionally represent the location of marginals with given dimensions can be helpful to distinguish LU-orbits. As e.g. shown in **Fig.** 6.3, it can distinguish more graphs than the rank lists can. However, for distinguishing entanglement classes, this is counterproductive. Indeed, because it represents the 'location' of the marginals, the marginal tensor T_k^G is not invariant under permutations of the nodes. This means that two elements from the same entanglement class may have different T_k^G, which makes them unsuitable as an identifier of entanglement classes. Indeed, **Fig.** 6.3 contains an example of this: the two graphs are permutations of each other and therefore from the same entanglement class, but their rank tensors T_2 are not equal.

On the other hand, the rank list *is* permutation invariant, and thus suitable as an identifier of entanglement classes. Still, it loses some information w.r.t. the marginal tensor, as it contains no information whatsoever regarding the (relative) positions of the marginals.

To circumvent this loss, it is desirable to obtain an identifier that is both permutation invariant and contains information regarding the relative locations. Such an identifier is formed by the *marginal eigenvalue*, which can be derived from the marginal tensor.

Definition 28. *For an n-node graph G with vertex set V, and a marginal size $k < n$, take its marginal tensor T_k^G as defined in Def. 27. Let H be the Hermitian $k \times k$ matrix obtained after summing T_k^G over $k-2$ arbitrary axes. Let $\{\lambda_1, \lambda_2, \ldots \lambda_k\}$ be the k real eigenvalues of H. Define the* marginal eigenvalue *t_k^G as the product of non-zero eigenvalues $\{\lambda_i\}$:*

$$t_k^G = \prod_{\lambda_i \neq 0} \lambda_i. \tag{6.17}$$

When context permits G is dropped, so that the marginal eigenvalue is written t_k.

Because eigenvalues are invariant under simultaneous permutation of the rows and columns of a matrix, t_k is permutation invariant. Moreover, it directly inherits its LU-invariance from T_k. Therefore, the marginal eigenvalue is indeed an identifier for entanglement classes.

Higher order marginals

The examples that have been shown so far all make use of the two-body marginal dimensions to show the LU-inequivalence of sets of graphs. This is not always adequate, as **Fig.** 6.4 shows: all two-body marginals of the two depicted graphs are maximally mixed, so that no conclusion can be made regarding their LU-equivalence. However, the highlighted three-body marginal, for which the two graphs have different marginal dimensions, shows that they are indeed LU-inequivalent.

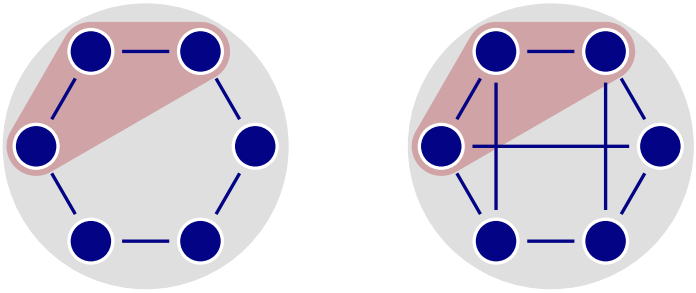


Figure 6.4: Two graph states $|G_1\rangle$ and $|G_2\rangle$ with the same dimension for all their two-body marginals, which are all trivial (i.e. maximally mixed). Nevertheless, the two graph states are LU-inequivalent, as the highlighted three-body marginal shows: that marginal for G_1 is non-trivial with $d_M = 1$, but the same marginal for G_2 is trivial.

Thus, it can be necessary to increase the marginal size k to determine LU-inequivalence of a set of graphs. A higher k is computationally more intensive (the number of k-body marginals of an n-qubit state is super-exponential in k, see also sec. 6.6), so it is not always preferred to increase the marginal size. It is thus an important question how well the identifiers perform w.r.t. the choice of k; this will be studied in sec. 6.4.

It should be noted that there is a limit after which increasing k cannot provide new insights. By (2.28), the marginal rank of any marginal M is directly determined by the marginal rank of its counterpart $M^\perp$. It follows that calculating T_k, l_k or t_k for any $k > \lfloor \frac{n}{2} \rfloor$ is superfluous: it is completely determined by its respective counterpart of marginal size $k' = n - k$, which is easier to calculate.

6.4 | Performance of the identifiers

It is the goal to assess how well the identifiers perform in their task of categorizing graphs into LU-orbits or entanglement classes. Because the marginal tensor T_k is the only identifier that is not invariant under permutations, it is the only identifier that is tested in its power to distinguish LU-orbits. The marginal list l_k and marginal eigenvalue t_k are tested in their power to distinguish entanglement classes.

Two figures of merit are used to assess the performance of the identifiers. These figures of merit can be computed for graphs and marginals of any size n or k, and are detailed as follows:

- The ratio of the number of different identifiers to the total number of LU-orbits (for T_k) or entanglement classes (for l_k and t_k); this ratio is denoted $r(T_k)$, $r(l_k)$ or $r(t_k)$.
- The probability that two random labelled (for T_k) or unlabelled (for l_k and t_k) LU-inequivalent graph states are evaluated to have identical identifiers; this probability is denoted $p(T_k)$, $p(l_k)$ or $p(t_k)$.

The first figure of merit reflects how well the identifiers perform to label different orbits or classes. If one wants to categorize all LU-orbits or entanglement classes, it is exceedingly useful to obtain a unique identifier for every one of these orbits. The ratio $r(\cdot)$ reflects how many different identifiers there are. In general, it holds that $0 < r(\cdot) \leqslant 1$, where a ratio of 1 indicates that every LU-orbit or entanglement class is labelled with a unique value for the identifier. A ratio that approaches zero indicates that the identifier fails to label any LU-orbit or entanglement class uniquely. As such, the performance of the identifier is better for higher ratios $r(\cdot)$, because then there are more unique labels.

The second figure of merit focusses more on graph states themselves. Consider, for example, the situation where graphs with equal identifiers are always assumed to be from the same LU-orbit. In such a setting, if different LU-orbits have identical values for the identifier T_k, this would inadvertently lead to incorrect conclusions. However, if this equal labelling happens only for two comparatively small LU-orbits, the fact that they are labelled with the same identifier is a relatively minor issue. It is then extremely likely that the mutual LU-(in)equivalence of two random graphs can faithfully be determined, since in such a case there are only a few cases that lead to false positives.

In general, it holds that $1 \geqslant p(\cdot) \geqslant 0$, where a probability of 0 indicates that any two graphs that are LU-inequivalent or from different entanglement classes will always be correctly distinguished. This means that a faithful decision on LU-(in)equivalence can always be taken. On the other hand, if the probability approaches 1, it means that no two graphs that belong to different

LU-orbits or entanglement classes can be distinguished. Thus, the performance of the identifier is better for lower probabilities $p(\cdot)$, as inconclusive results or false positives are then less likely to occur.

The test of performance is facilitated by an online database of every local complementation orbit of graphs up to 9 qubits, which is provided as supplementary material of [115] and can be found at [136]. The database contains a representative of every LC-class (see sec. 4.3), which means that two points need to be taken into consideration:

- If the database is used as-is to test for LU-equivalence, essentially it is assumed that the LU-LC conjecture is true for all graphs in the database. As noted before (see sec. 4.4.2), the conjecture is indeed true for all entanglement classes up to 8 qubits [116, 138], but special care is warranted for larger graphs or orbits.

- Not considering the above point, the database provides representatives of entanglement classes. To test the performance of T_k (i.e. for distinguishing LU-orbits) representatives of every LU-orbit need to be obtained. These need to be computed from the representative of the entanglement class by calculating all its permutations and categorizing those into separate groups of LU-orbits, as explained in sec. 4.3.

6.4.1 | Performance for LU-orbits

The two figures of merit are calculated for the marginal tensor T_k for all LU-orbits[2] of size $3 \leqslant n \leqslant 8$ and of all LC-orbits of size 9, and marginals of size $2 \leqslant k \leqslant \lfloor \frac{n}{2} \rfloor$. The results are shown in **Tab.** 6.1.

For small graphs, identifying the two-body marginal dimensions (i.e. taking $k = 2$) provides enough information to uniquely label all LU-orbits - at the same time increasing k would not provide any extra insights, as explained at the end of sec. 6.3 (see additionally (2.28)). For $n = 6$, which is the lowest graph size for which the three-body marginals are independent from the two-body marginals, T_2 proves considerably less effective than T_3. This behaviour is the same for larger graphs as well: increasing k always provides a higher ratio and a lower probability.

Moreover, for graphs up to 8 qubits, there always exists a marginal size k such that the identifier works flawlessly: the ratio reaches 1, indicating that every LU-orbit has a unique value for the identifier, and the probability reaches 0, indicating that every pair of LU-inequivalent graphs can be distinguished.

At the same time, for a fixed marginal size k the performance deteriorates drastically as n increases. Only identifying two-body marginals, for 6 nodes

[2] Actually, the database technically covers LC-orbits instead of LU-orbits. [116] showed only that the LU-LC conjecture is true LC-classes and entanglement classes, but the results of this section show that it holds for LU- and LC-orbits as well. This means that LC- and LU-orbits of graphs up to and including 8 nodes are indeed identical.

n	$r(T_2)$	$r(T_3)$	$r(T_4)$	$p(T_2)$	$p(T_3)$	$p(T_4)$
3	1	-	-	0	-	-
4	1	-	-	0	-	-
5	1	-	-	0	-	-
6	0.52	1	-	0.05	0	-
7	0.13	1	-	0.12	0	-
8	0.02	0.88	1	0.22	0.0001	0
9	0.001	0.48	0.999	0.37	0.0004	3e-10

Table 6.1: The performance of T_k as an identifier of LU-orbits is tested by computing the figures of merit $r(T_k)$ and $p(T_k)$ for all marginal sizes $2 \leqslant k \leqslant \lfloor \frac{n}{2} \rfloor$ and all LU- (or LC-)orbits of size $3 \leqslant n \leqslant 9$. If $r(T_k) = 1$, every LU-orbit has a unique T_k, which can thus serve as a unique identifier for the orbit. If $p(T_k) = 0$, two random LU-inequivalent graphs will always be distinguished by their marginal tensor. Perfect labelling is obtained for all graphs of size $n \leqslant 8$, provided a large enough k is used. For 9 qubits no identifier has this desirable property. Indeed, LU-inequivalent graphs exist that have the exact same structure for their marginal dimensions; sec. 6.5 addresses these in more detail.

the number of unique labels is only roughly half the number of LU-orbits (consider e.g. **Fig.** 6.4); for 9 nodes this is reduced to about just 0.1%.

Interestingly, this effect is less pronounced for the probabilities. Even though for $n = 6$, $k = 2$, there are only about half as many unique labels as there are LU-orbits, the probability of obtaining a false positive (i.e. assuming that two LU-inequivalent graphs with the same identifier T_2 are LU-equivalent) is only 5%; even for graphs of 9 nodes the probability is well under 50%.

Increasing the marginal size to $k = 3$ increases the performance considerably: T_3 performs perfectly for graphs of size 6 and 7, and false positives are extremely unlikely for 8 and even 9 nodes. Nevertheless, marginals of size $k = 4$ are needed to correctly identify and label *all* 8-node LU-orbits.

For $n = 9$, the marginal tensor fails to perform perfectly even for $k = 4$; the reason for this is that there are two different orbits with the exact same structure for their marginal dimensions, so that their marginal tensors inevitably are identical as well. Representatives of both orbits are shown in **Fig.** 6.7. Interestingly, the cut-off between perfect and imperfect performance lies exactly at the boundary of the lower bound to the LU-LC conjecture [116, 138].

As explained above, the graphs in **Fig.** 6.7 are two representatives of *LC-orbits* rather than LU-orbits; it is thus known that these two representatives cannot be distinguished by their marginal tensors, even though they

are LC-inequivalent. This makes for an interesting pair of graphs, as they are LC-inequivalent without a conclusive answer regarding their LU-equivalence. In other words, the pair forms a potential counterexample to the LU-LC conjecture. This pair is the only such candidate for $n = 9$, and their LU-(in)equivalence is addressed in more detail in sec. 6.5.

6.4.2 | Performance for entanglement classes

Similar to the LU-orbits, the two figures of merit are calculated for every entanglement class of size $3 \leqslant n \leqslant 8$ and LC-class of size 9, and for marginals of size $2 \leqslant k \leqslant \lfloor \frac{n}{2} \rfloor$. The figures of merit are calculated for both l_k and t_k. The results for l_k are listed in **Tab.** 6.2, and the results for t_k are listed in **Tab.** 6.3.

n	$r(l_2)$	$r(l_3)$	$r(l_4)$	**R**	$p(l_2)$	$p(l_3)$	$p(l_4)$	**P**
3	1	-	-	1	0	-	-	0
4	1	-	-	1	0	-	-	0
5	1	-	-	1	0	-	-	0
6	0.73	0.82	-	1	0.01	0.01	-	0
7	0.42	0.85	-	0.92	0.17	0.03	-	0.03
8	0.15	0.54	0.56	0.94	0.30	0.05	0.03	0.01
9	0.04	0.34	0.70	0.83	0.44	0.05	0.01	0.01

Table 6.2: The performance of l_k as an identifier is tested by computing the figures of merit $r(l_k)$ and $p(l_k)$ for all marginal sizes $2 \leqslant k \leqslant \lfloor \frac{n}{2} \rfloor$ and all entanglement classes of size $3 \leqslant n \leqslant 9$. Note that for $n = 9$ the set of LC-classes (see sec. 4.3) is used, as explained at the start of sec. 6.4. These are not necessarily exactly the same as the entanglement classes, which concern LU-equivalence. If $r(l_k) = 1$, every entanglement class has a unique l_k, which can thus serve as a perfect identifier for the class. If $p(l_k) = 0$, graphs from two randomly selected but different entanglement classes are always correctly distinguished. The columns **R** and **P** detail the ratio and probability when the identifiers for every different k are combined.

As with the LU-orbits, increasing the marginal size k for a fixed n always provides better results. Still, perfect results are only obtained for entanglement classes up to 6 nodes; entanglement classes of larger graphs can never be perfectly labelled for any single k.

Fixing k and increasing n again shows quick deterioration: the ratio $r(l_2)$ goes from 1 ($n = 5$) to 0.73 ($n = 6$) down to just 0.04 ($n = 9$). Similar behaviour exists for l_3, but $r(l_4)$ is actually *higher* for $n = 9$ than for $n = 8$. This is most likely an oddity w.r.t. the 8-node graphs.

Similarly to the LU-orbits, the probabilities show the same behaviour as the ratios, although with less severe deterioration. The probability of a false positive for any graph size never exceeds 3%, provided a large enough k is used.

Fixing $k = 3$, even for entanglement classes of size 8 and 9 the probability never exceeds 5%. Moreover, the discrepancy with the ratio of l_4 being better for 9 nodes than for 8 nodes can not be found for the probabilities.

Interestingly, combining the results for multiple marginal sizes k (i.e. the columns marked **R** and **P**) can provide more insights than any individual k alone. Consider for example the ratios: the 6-qubit entanglement classes cannot be labelled perfectly by using either l_2 or l_3. However, when both are used as a label at the same time, the ratio becomes 1, meaning that every entanglement class is uniquely identified. Similar behaviour is shown for larger graphs, although here no perfect labelling is retrieved by combining the different marginal sizes. The combination of multiple marginal sizes can provide an improvement for the probabilities as well. However, here the effect is far less pronounced: only for the entanglement classes of 8 nodes the combination actually provides a better result.

This means that there are pairs of entanglement classes that have the exact same 3-body marginal dimensions, even though at least one 2-body marginal differs in dimension. An example of such a pair is given in **Fig.** 6.5: l_3 is equal for both graphs, but l_2 is able to distinguish them. This shows that focusing solely on larger k can be counterproductive.

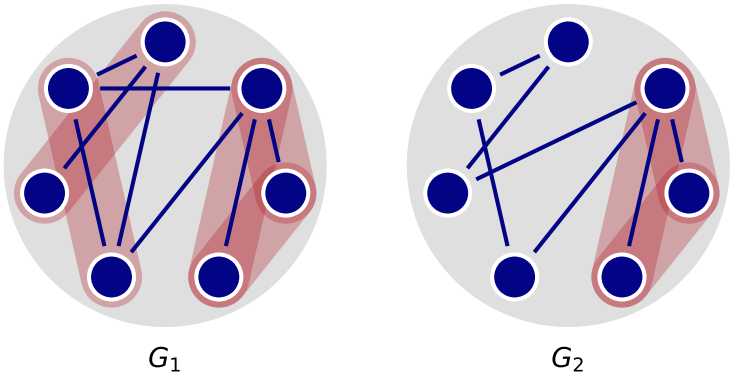


Figure 6.5: Two graph states $|G_1\rangle$ and $|G_2\rangle$ that are from different entanglement classes, which can be shown by comparing their marginal lists. Indeed, for $k = 2$ the lists are different: $l_2^{G_1} = [16, 5, 0]$ but $l_2^{G_2} = [18, 3, 0]$. All non-trivial two-body marginals have been highlighted; since they differ in number, the inequivalence of the two graphs follows. Interestingly, their marginal lists for $k = 3$ coincide: $l_3^{G_1} = l_3^{G_2} = [12, 22, 1, 0]$, so that their inequivalence cannot be determined from the three-body marginals. This shows that a higher k does *not* always offer a better or even equally performing identifier, but can be counterproductive.

Additionally, a perhaps unexpected phenomenon occurs. The marginal

lists are able to provide a *larger* ratio for the 8-node graphs than for the 7-node graphs. This is perhaps, at least in part, explained by the fact that combining the marginal lists of different k is especially effective for 8 nodes: for no single k the ratio exceeds 0.56, but combining all marginal lists obtains a ratio of 0.94.

For the probabilities a similar phenomenon occurs, not only for the 8 node graphs, but for the 9 node graphs as well. Indeed, a lower probability of false positives can be obtained for both these sizes compared to the 7 node graphs.

n	$r(t_2)$	$r(t_3)$	$r(t_4)$	**R**	$p(t_2)$	$p(t_3)$	$p(t_4)$	**P**
3	1	-	-	1	0	-	-	0
4	1	-	-	1	0	-	-	0
5	1	-	-	1	0	-	-	0
6	0.73	1	-	1	0.01	0	-	0
7	0.46	1	-	1	0.16	0	-	0
8	0.19	0.89	1	1	0.30	0.0001	0	0
9	0.06	0.73	0.998	0.998	0.44	0.01	1e-06	1e-06

Table 6.3: Similarly to l_k in **Tab.** 6.2, the performance of t_k as an identifier is tested by computing the figures of merit $r(t_k)$ and $p(t_k)$ for all marginal sizes $2 \leqslant k \leqslant \lfloor \frac{n}{2} \rfloor$ and all entanglement classes of size $3 \leqslant n \leqslant 9$ (with the caveat that for $n = 9$ the set of LC-classes is used, as explained at the start of sec. 6.4). If $r(t_k) = 0$, every entanglement class has a unique t_k, which can serve as a unique identifier for the class. If $p(t_k) = 0$, graphs from two randomly selected but different entanglement classes are always correctly distinguished.

In general, the behaviour for the marginal eigenvalues t_k is similar to that for l_k, but overall it performs better than the lists. Most importantly, the overall behaviour of T_k for LU-orbits (i.e. **Tab.** 6.1) is retrieved. In particular this means that up to 8 nodes there is always a k that perfectly distinguishes all entanglement classes. An example of a set of entanglement classes that cannot be distinguished by their marginal lists l_k but can be by their marginal eigenvalue t_k can be found in **Fig.** 6.6.

Additional similar behaviour is apparent: fixing k, the performance deteriorates quickly with increasing n. Again, this is more severe for the ratios than for the probabilities. For e.g. $n = 9$, the ratio has dropped down to $r(t_2) = 0.06$, but there is still less than a 50% chance that a false positive occurs. In general, the probabilities for a false positive are exceedingly small, or zero, for large enough k.

Again for 9 nodes there is no perfect labelling possible, similar to the case for T_k with LU-orbits. This shows that the two aforementioned graphs with the exact same structure for their marginal dimensions (i.e. those shown in **Fig.** 6.7 and addressed in more detail in sec. 6.5) are not just LC-inequivalent,

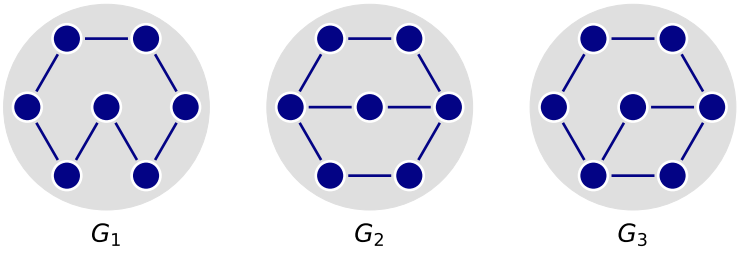


Figure 6.6: Representatives $|G_1\rangle, |G_2\rangle$ and $|G_3\rangle$ of three different entanglement classes. It holds that $l_k^{G_1} = l_k^{G_2} = l_k^{G_3}$, so that the marginal lists are not able to distinguish the classes. However, $t_3^{G_1}, t_3^{G_2}$ and $t_3^{G_3}$ are all different, so that the marginal eigenvalue can be used to distinguish the classes. This shows that sometimes a marginal eigenvalue is able to distinguish classes that no marginal list can.

but they belong to different LC-classes (see sec. 4.3) as well.

Contrary to the marginal lists, combining identifiers of different marginal sizes k does *not* provide any extra information compared to individual t_k's, which is the same behaviour as for the marginal tensor. For a marginal tensor of size k, the marginal tensor of lower size $k' < k$ is 'embedded' into T_k, which means that no information can be lost by increasing k. Due to the nature of how it is computed, t_k could theoretically have this flaw, but as is evident from the results it does not occur.

6.5 Different LC-orbits with equal identifiers

Tabs. 6.1 and 6.3 show that, as explained before, there are separate LC-classes and LC-orbits that have the same structure for their marginal dimensions. This means that they cannot be distinguished by T_k or t_k, respectively, for any marginal size k. **Fig.** 6.7 shows two graphs L and R, whose associated graph states are representatives for two different LC-orbits $\mathcal{O}^{\mathrm{LC}}(|L\rangle)$ and $\mathcal{O}^{\mathrm{LC}}(|R\rangle)$ that have the same structure for their marginal dimensions, i.e. $T_k^L = T_k^R$ for every k. They form the smallest example of such a pair of LC-orbits, and the only nine-qubit example.

The identifiers are invariants for LU-operations, but the two graphs are known to be from different LC-orbits. Therefore, it is in principle possible that $|L\rangle$ and $|R\rangle$ are LU-equivalent, even though they are LC-inequivalent,. This would make them the smallest counterexample to the LU-LC conjecture, because it holds true for at least all graph states up to 8 qubits (see [116] for entanglement classes, or **Tab.** 6.1 for LU-orbits).

However, it indeed holds $|L\rangle$ and $|R\rangle$ are LU-inequivalent as well, which is the statement of Lemma 12 from Pub. **[G]** ([55]). For a proof the reader

is referred to that publication. This shows that there are LU-orbits and entanglement classes that cannot be distinguished by the structure of their marginal dimensions, and therefore by any of the identifiers presented in this chapter.

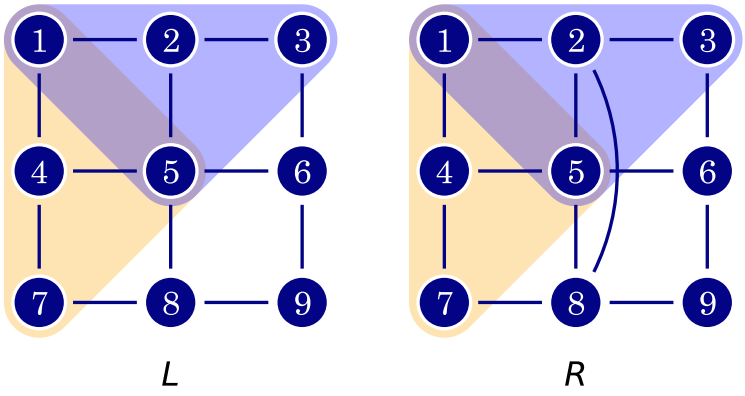


Figure 6.7: Two graph states $|L\rangle$ and $|R\rangle$ that are LC-inequivalent. Still, the dimensions of all of their marginals align, so that their T_k's are the same for every k. This means that no conclusion can be made regarding their LU-inequivalence from their marginal tensors. Therefore, they form a potential counterexample to the LU-LC conjecture. However, in Pub. **[G]** ([55]) it is shown that the graphs are indeed LU-inequivalent.

As a final interesting note, there are pairs of *isomorphic* graphs that cannot be distinguished by their marginal tensors for any k, but are still LU-inequivalent. This shows that there are separate LU-orbits from a single entanglement class with the exact same structure for their marginal dimensions. One example is formed by the *Peterson* graph, which can be found in **Fig.** 6.8, also presented in Pub. **[G]** ([55]). It holds that $\mathcal{E}_{\mathcal{C}}(|P\rangle) = \mathcal{E}_{\mathcal{C}}(\left|\tilde{P}\right\rangle)$, but $\mathcal{O}^{\mathrm{LU}}(|P\rangle) \neq \mathcal{O}^{\mathrm{LU}}(\left|\tilde{P}\right\rangle)$, even though $T_k^P = T_k^{\tilde{P}}$ for every k.

6.6 | Efficiency of the introduced methods

It is exponentially hard to compute the marginal for an arbitrary quantum state. However, the method presented in sec. 6.1 can be used to compute the stabilizer dimension of any k-body marginal by calculating the nullity of the $(n-k) \times (k)$ binary matrix $\Gamma^{(M^\perp, M)}$. This can be performed by Gaussian elimination over $\mathbb{F}_2$, which has a complexity of $\mathcal{O}((n-k)k^2)$ or $\mathcal{O}((n-k)^2k)$, whichever is lowest. Because calculating the marginal dimension of any marginal with $k > \lfloor \frac{n}{2} \rfloor$ is not relevant (see sec. 6.3), the complexity of calculating d_M is $\mathcal{O}((n-k)k^2)$.

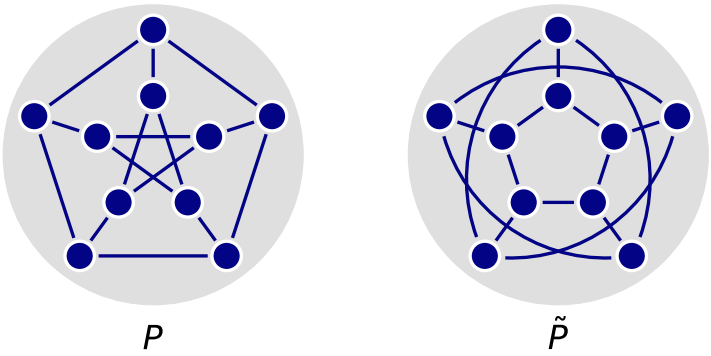


Figure 6.8: The two graphs P (the *Peterson graph*) and $\tilde{P}$ are isomorphic: permuting all 'inner' with all 'outer' nodes of one graph results in the other. Hence, the graph states $|P\rangle$ and $\left|\tilde{P}\right\rangle$ are part of the same entanglement class. Moreover, beyond their marginal lists l_k and eigenvalues t_k, their marginal tensors T_k coincide as well. However, they are not LU-equivalent; this shows that there are cases were different LU-orbits cannot be distinguished by the methods presented in this chapter, even though they are part of the same entanglement class.

For a given graph size n and marginal size k, there are $\binom{n}{k} = \frac{n!}{(n-k)!\cdot k!}$ marginals. This means that, using Sterling's approximation, calculating the dimensions d_M of every M with a fixed size k is $\mathcal{O}(k^{\frac{3}{2}-k}n^{k+1})$. It follows that the complexity of calculating l_k or T_k of an n-node graph G for $k \geqslant 2$ is of that same order.

For t_k, the eigenvalues of an $n \times n$ Hermitian matrix have to be calculated. Although there technically exist bounds that are lower, in practice this is $\mathcal{O}(n^3)$. Thus, calculating t_k is $\mathcal{O}(k^{\frac{3}{2}-k}n^{k+1})$, or $\mathcal{O}(n^3)$ if given access to T_k.

6.7 Conclusion

The methods that are presented in this chapter can be used to inspect the local equivalence of graph states, and by extension stabilizer states. They consider LU-equivalence, so that the methods are more versatile than the results that focus solely on LC-equivalence that were presented in chapter 4.

It should be noted that recent work, Pub. **[H]** ([117]), introduces a new method to verify the LU-equivalence of graph states. It is an algorithm that works similar to the Bouchet algorithm for LC-equivalence (see sec. 4.4): it takes two graphs as input, and outputs either `NO` if the two graphs are not LU equivalent, or outputs an exact form of the local unitary operator under which they are equivalent. The first step of the algorithm involves an inspection of

the marginals dimensions such as presented in this chapter. A subsequent step of the algorithm is exactly the method used in Pub. **[G]** to show that the two graphs from **Fig.** 6.7 are LU-inequivalent.

This chapter has focussed on equivalence of graph states solely under local unitary operations. Although e.g. chapter 5 addresses the more general setting that includes measurements, it does this for a very specific set of resource and target states. The effect of measurements on the stabilizer dimension can be studied, so that potentially the methods presented in this chapter can be adapted to assess the equivalence of graphs when node deletions are included.

Part II has introduced and discussed, through chapters 4 to 6, various methods to study, characterise and manipulate multi-partite entanglement in networks. Part III, the next part of this thesis, aims to discuss the utilization of multi-partite entanglement in quantum networking and cryptography. More specifically, the usage of multi-partite entanglement is studied in networking tasks where *anonymity* must be guaranteed. In such anonymous settings, the identity of the nodes of a network that are involved in the networking protocols must remain hidden from other parties.

PART III

Anonymous Conference Key Agreement

7

Introduction to Quantum Cryptography

In part II the question was addressed how to manipulate or obtain different forms of entanglement in quantum networks, with a focus on multi-partite entangled states. Part III takes a turn to a more operational topic, by studying how multi-partite entanglement can be used in networking protocols to realise cryptographic tasks. Specifically, the topic of *anonymous conference key agreement* (ACKA) is discussed in chapters 8 to 10.

Conference key agreement (*CKA*) is a generalisation of the well-known topic of *quantum key distribution* (*QKD*) to more than two parties. QKD provides a method for two parties in a network to establish a secret key that can be used for cryptographic tasks.

The topic of *anonymity* in quantum networking is relatively new. Here, the goal is not to hide e.g. the content of a message, but rather the identity of the source, or recipient, or both. As such, it can be interpreted as an addendum to the requirements of a cryptographic protocol: ACKA aims to provide conference key agreement, with the added requirement that the involved parties remain anonymous.

This chapter introduces the concepts regarding QKD that are applicable to this thesis. For completeness, the basics of cryptography that are relevant to this thesis are discussed first, in sec. 7.1. In sec. 7.2, the fundamental concepts of QKD are introduced, and an intuition behind its functionalities and security is discussed.

Modern QKD thrives because of rigorous security definitions, that cover both the *secrecy* and *correctness* of the generated key. Secrecy indicates that the key is only known to the parties that are communicating, colloquially referred to as *Alice* and *Bob*. Correctness indicates that the keys that Alice and Bob generate are identical. Section 7.3 provides this rigorous security definition.

An integral part of any QKD protocol to obtain security is *post-processing*, which can be understood as a collection of steps to perform *error correction*, providing correctness, and *privacy amplification*, providing secrecy. These post-processing steps, and how they imply security through *security proofs*, are addressed in sec. 7.3 as well.

Conference key agreement, the generalization of QKD to more than two parties, is addressed in sec. 7.4. Subsequently, the concept of *anonymity* is introduced in sec. 7.5, which specifically discusses *anonymous conference key agreement* (ACKA), the topic of chapters 8 to 10 that are based on Pubs. **[A]** to **[E]**. The chapter is concluded in sec. 7.6.

The contents of this chapter are largely an introduction based on literature [7, 17, 72], with the exception of sec. 7.5, whose contents (including the definitions of anonymity) were originally presented in Pubs. **[A]** and **[C]**.

7.1 Basics of cryptography

Cryptography is a field of research with a wide range of applications. The original and most widespread application is that of *private* communication, which this chapter will mostly focus on. More specifically, the setting has two parties that are colloquially known as *Alice* and *Bob* but also referred to as A and B[1]. Alice and Bob want to communicate: they want to exchange a message, usually assumed to be sent by Alice to Bob. Alice and Bob are physically separated from each other, but can communicate over e.g. the internet. Either way, they cannot prevent anyone from inspecting their messages, reading and potentially copying them. Complementing the names Alice and Bob, this *eavesdropper* is known as Eve, or just E, and is additionally referred to as the *adversary*. The means of communication that allow Alice to send a message to Bob is referred to as a *channel*. Since any potential adversary is assumed to be able to read the contents of the message, the channel is said to be a *public channel*.

To prevent Eve from being able to read the message Alice *encrypts* the message m, resulting in a *cyphertext* c. Instead of sending m she sends c over the public channel: Eve then has access only to c, but not to m. Provided Alice and Bob use a good encryption scheme, m cannot 'reasonably'[2] be obtained

[1] A and B is used to refer to the quantum systems of A and B as well, if they have one.
[2] What is meant by *reasonably* is addressed later in this section.

from c, so that Eve is not able to read the message. Bob, upon receiving c, *decrypts* the cyphertext and recovers the original message m.

It is then the question why Bob would be able to decrypt the message, whereas Eve is not able to do so. To obtain this desired effect, Alice and Bob need to use a shared secret k, called the *key*. The encryption and decryption both use k, so that without it decryption is not possible. This means that the key needs to be strictly secret and only shared between Alice and Bob. There are two solutions to realise this:

- Before the protocol runs, Alice and Bob meet and agree on a secret key, or they use a trusted courier (i.e. a 'private' channel). This is usually referred to as a *pre-shared secret*.
- Alice and Bob perform *public key exchange* [139], so that they can agree on a secret key. The most well-known examples of public key cryptography are Diffie-Hellman key exchange [57] and the RSA cryptosystem [58].

The first option is not always applicable or practical, but the second option is only possible using *assumptions* on the power of the adversary. More specifically, the key exchange process is facilitated by calculations that are straightforward to perform in one direction, but are hard to invert. The assumption is then that these calculations are too hard to perform for the adversary, and therefore these assumptions are called *computational assumptions*. The most well-known example of such a *one-way function* is calculating the product of two co-prime numbers, which comes from the RSA cryptosystem. Computing products of two numbers is straightforward with simple computers, but *factoring* to retrieve the two original co-prime numbers from their product is increasingly hard. This task generalizes to the *discrete logarithm* problem, which additionally is the relevant one-way function of most Diffie-Hellman key exchanges.

Factorization and solving the discrete logarithm problem is classically hard to perform, but there exist efficient quantum algorithms to perform these tasks - most notably the celebrated Shor's algorithm [61, 62]. Hence, with the recent advent of (rudimentary) quantum computers [59, 60], there is a need for different methods to obtain shared secret keys.

Quantum Key Distribution (QKD) aims to provide shared secret keys by the communication of quantum signals. In principle, this approach allows for *information-theoretic* security: no assumptions or restrictions are put on the adversary, but the security of the process comes from the laws of physics. Providing that our understanding of nature is correct, QKD can provide unconditional security[3].

[3] An implicit assumption that is still present is that the communication channels are *authenticated*, which will be discussed shortly.

7.1.1 | Security of the key

Regardless of what method Alice and Bob use to obtain a shared secret key, it results in Alice having a key k_A and Bob having a key k_B. In the ideal case $k_A = k_B = k$, but in an imperfect scenario the two keys will not be identical. Still, the keys need to be strongly *correlated* (see sec. 1.4): knowing k_A gives a lot or all information regarding k_B, and vice versa. At the same time, k_A and k_B must be as *uncorrelated* as possible with whatever information Eve has access to. This information is referred to as Eve's *side information*, and includes all communication over public channels and any quantum registers that Eve may have. Non-perfect cases, which are inevitable in the real world, need to be carefully treated. Under such careful treatment, even when the keys are not completely correlated with each other, or completely uncorrelated with Eve's side information, security can be obtained. An intuition why QKD provides security is discussed in sec. 7.2, while sec. 7.3 addresses security and how to prove it in more detail.

7.1.2 | Authenticated channels

The concept of public channels allows anyone to see the contents of a communicated message, but an important assumption on the channel usually remains: it is often the case that the channel is *authenticated*, which means that Bob knows that the message came from Alice. Such an authenticated channel is usually necessary to perform many cryptographic tasks, including QKD. To realize an authenticated channel is not a trivial task, but the RSA cryptosystem can be used to implement one. However, this would nullify the purpose of QKD, as RSA resides on computational assumptions. QKD, aiming to provide unconditional security, would then inevitably make use of RSA, which it is trying to replace.

The security and effectiveness of QKD is debated mostly regarding the topic of channel authentication. A somewhat unsatisfying solution is offered by the perspective that, even if assumptions must be made, these assumptions must hold only *during* the QKD-process, resulting in ever-lasting security afterwards. This is in contrast to purely classical methods: there, all encrypted communication can be copied and stored during transmission, and cracked and decrypted later[4].

Another approach is to utilize a shared secret between Alice and Bob to realize an information-theoretic authenticated public channel. If the key that is created is longer than what was initially necessary to realize the authenticated channel, there is a net positive amount of key that comes out of the protocol. Repeated rounds of QKD can then realize an arbitrary amount of secret key. In this sense, QKD is a *key-growing* or *key-expanding* protocol. This is the approach taken in this thesis as well: Alice and Bob are always

[4] An adversary model aptly named *store-now-decrypt-later*. It is believed to be widely used currently by many larger parties, with the promise of quantum computers on the horizon to eventually break the current encryption.

assumed to have a shared initial secret, from which they can obtain *more* information-theoretically secure key.

7.2 | Introduction to quantum key distribution

QKD is not a single set of rules, but a conglomeration of different techniques and methods that all involve quantum communication to realize secure and secret key. `BB84` [6], the first proposed QKD protocol, is a *prepare-and-send* protocol [17]: in such protocols, Alice prepares a quantum state and sends this (over a public *quantum* channel) to Bob. Other prepare-and-send protocols include the *six state protocol* [140] and the `B92` protocol [141].

Modern security proofs [1] obtain security for prepare-and-send protocols by a reduction to equivalent *entanglement-based* protocols [17], that utilize entangled states and the non-classical correlations that they can provide to obtain security. The most well-known entanglement-based protocols are `E91` [142], which essentially performs a Bell test [34, 83] to guarantee that any adversary must be uncorrelated with Alice and Bob, and `BB92` [143], which is closely related to `BB84` and does *not* involve a Bell test. `BB92` is easier to implement than `E91`, but `E91` has a stronger security guarantee. In particular, due to the Bell test, less trust needs to be put in the hardware that Alice and Bob use to implement the protocol. As such, `E91` can be seen as the first *device independent* QKD (*DI-QKD*) protocol [144], where (at least some of) the hardware of Alice and Bob is treated as a black box. Eve is assumed to have full power over this hardware. Surprisingly, security is still possible in such a scenario, but performance is often detrimentally affected in comparison with QKD protocols that are not device independent.

There exist myriad other, different QKD protocols [17] that function in many distinct ways, but a complete overview is beyond the scope of this thesis. Instead, this introduction focusses on entanglement-based protocols only.

7.2.1 | The basics steps of a QKD protocol

Any QKD protocol can be understood to consist of four or five separate parts, of which only the first actually involves quantum communication. The other steps are considered *post-processing* and are purely classical, but do involve (classical) communication. In the first step, one or more quantum states are prepared, communicated as quantum signals and subsequently measured. The communication is not necessarily from Alice to Bob: the direction could be reversed, and it is also common that a third party is involved, who creates the quantum states that Alice and Bob receive. Additionally, there are *two-way* QKD protocols, in which the quantum signals are being sent forth-and-back. These are addressed briefly in sec. 7.2.2.

After the communication, Alice, Bob or both perform some measurement resulting in an outcome on their respective quantum systems A and B; these outcomes form the basis of the ultimate key. The choice of measurement basis

is usually random for every party that measures, and it needs to be recorded. This first step is repeated many times, so that enough key can be created. This concludes the quantum part of the protocol.

The second step, called *sifting*, involves discarding part of the generated measurement results. Sifting does not occur in every protocol, but is necessary for every round where Alice and Bob used incompatible measurement bases - what is considered incompatible is dictated by the specific protocol. What is left over is known as the *raw key*, with length L, referred to as the *block size*.

Due to the presence of noise in the quantum channel and a (potential) adversary that is interfering, the sifted raw keys of Alice and Bob might not be identical or secret. Integral to QKD is that active interference of an adversary would always result in a certain type of noise on the raw key, which can be estimated. This noise needs to be accounted for in later steps, but these steps reduce the length of the raw key - the worse the noise, the more raw key has to be forfeited. If the amount of noise reaches a certain threshold, it is assumed that Eve has interfered to such an extent that no secret key can be created. It is therefore vital that the noise levels are estimated and compared against pre-determined thresholds - this step is known as *parameter estimation*.

The noise that is *not* attributed to an adversary results in incorrect keys: Alice's and Bob's raw keys are not identical. This would render them unusable in any cryptographic application, so that the differences between the two raw keys have to be *corrected*. The step called *error correction* performs this, which involves some public communication between the two parties, and allows Bob to correct any errors his raw key might have w.r.t. Alice's raw key. Error correction involves another round of public communication, so that the success of the error correction can be verified.

Any noise that *can* be attributed to Eve, is assumed to be caused by her. The amount of correlation (measured in a suitable entropic measure, see sec. 1.4) that Eve can have with the raw key (identical for Alice and Bob after error correction), is directly computed from the amount of noise. This correlation with the raw keys is removed by Alice and Bob by *distilling* the secret key from the raw key. The final step, *privacy amplification*, provides this: it distils a secret key of length $\ell < L$ from the raw key.

The necessary reduction in key length by privacy amplification is upper bounded by the amount of side information that Eve can have. This includes the amount of correlation computed from the noise level, and additionally includes the amount of public communication during error correction.

To simplify analysis it is often assumed that $L \to \infty$. However, for finite L, *finite* effects are introduced in the parameter estimation, which additionally have to be accounted for. In this *finite regime*, the parameter estimation is imperfect, so special care in the security proofs needs to be taken. This can greatly complicate the analysis, especially for smaller block sizes L. Such *finite key effects* reduce the key length ℓ further, but their influence vanishes for larger block sizes. These finite key effects are largely determined by the fact

that, for smaller L, there is a larger ambiguity in the estimated parameters. To still provide security, this statistical uncertainty is taken in the 'worst-case' interpretation, so that the estimate of the noise level becomes much higher than its true level.

If the noise levels are too high, or the finite key effects are too strong (i.e. when L is too small), no distillation is possible with a non-negative secret key length. Increasing the block size always provides better parameter estimation, so that ℓ is (relatively) larger for larger L, even for fixed noise parameters. To emphasize the dependence on the block size, the secret key length ℓ will usually be written as a function of L in this thesis, i.e. $\ell(L)$. It is thus natural to consider the *ratio* of amount of secret key per block size L. This ratio is called[5] the *key rate* r, and is monotonically increasing with L and decreasing with the noise parameter.

In the setting that an infinite number of rounds have occurred, all the finite key effects have vanished. In this *asymptotic regime*, the key rate becomes the *asymptotic key rate* r_a:

$$r_a = \lim_{L\to\infty} \frac{\ell(L)}{L}. \tag{7.1}$$

For many QKD protocols, finite key effects become unimportant for a block size of $L \sim 10^9$, so that the finite and asymptotic key rates practically coincide.

7.2.2 | Other topics in QKD

There are various topics and details that have not been addressed in the previous sections, even though they warrant mentioning. They are listed here in arbitrary order.

Attenuated laser pulses and the *PNS* attack

Early QKD protocols like `BB84`, `E91` and `BB92` all assume that the quantum signals that are being communicated are true single-qubit states. Although a single qubit can be represented by a single photon, in practice it is extremely hard, if not impossible, to create a single photon and send it over macroscopic distances without it being lost. To circumvent this, an *attenuated laser pulse* can be used instead: a coherent bundle of single-wavelength photons that follow a Poisson distribution with an average photon number μ that is below one. Such beams of light are much easier to communicate over long distances, but they involve intricacies due to the fact that the signal is not represented by a single qubit any more. More specifically, even though the *average* number of photons μ is below one, there is a finite chance that multiple photons exist in the channel, which can create problems for security. The most well-known

[5] To call this the key rate is a convention used by theorists. Experimentalists, on the other hand, might use the term 'key rate' for something else, namely roughly the number of generated raw key bits per 'channel uses' (e.g. laser pulses). These two notions are incompatible.

such problem is the *photon number splitting* or PNS attack [145, 146], where Eve 'snoops' any extra photon that might exist in the channel. Without alteration of the underlying protocol, this has strong implications on security and drastically affects the key rates, but methods exist to remedy this attack.

Indeed, by sending *decoy states* [147–149] this attack can be mitigated. Instead of sending the normal pulse with average photon number μ, a decoy state pulse can be randomly sent. This decoy state pulse has an average photon number randomly chosen from a pre-determined set of values; only after all communication has happened the average photon numbers are communicated. Other approaches exist, like the `SARG04` protocol [150], which circumvents the PNS attack by not directly communicating the measurement bases (necessary for sifting), but encoding them in two non-orthogonal quantum states (much like the `B92` protocol).

Another approach to solve the PNS attack is to not use an attenuated laser as the quantum signal source, but instead use a source that is able to create a photon distribution that is much more close to a true single-photon distribution. Most notable are the *quantum dots* [151], which provide a distribution that results, at least in theory, in stronger key rates than decoy-state methods.

Continuous variable QKD

Modern security proofs [1] don't assume that the transmitted signals are single qubits, but model the quantum states in Hilbert spaces of arbitrary (but finite) size. This means that the signal is still *discrete* (i.e. a superposition of a discrete number of basis states), so that these types of protocols are known as *discrete variable*- (DV-) protocols [17].

In comparison, *continuous variable*- (CV-) protocols [17, 152–154] use continuous signals in infinite Hilbert spaces [155, 156]. Such CV protocols are less prone to noise and have a higher theoretical limit on the key rate per channel use. However, even though security proofs exist [157, 158], both error correction and finite key effects are much harder to address, which makes CV-QKD less practical with current technologies.

Two-way QKD

All protocols that have been named involve Alice sending a (quantum) signal to Bob, or a third party distributing entanglement between the two parties. These protocols are known as *one-way* protocols, as they involve signals going from one location to another, but never back over the same channel.

In contrast, *two-way* QKD protocols [159, 160] involve multiple rounds of quantum communication back and forth between Alice and Bob. In a two-way protocol, Bob applies a unitary transformations to the signal before sending it back, instead of immediately measuring it. Such protocols allow, in principle, for high key rates, but are greatly affected by noise. Moreover, due to their

two-way nature and because they involve unitary operations on the signal, they are considerably harder to implement than most one-way protocols.

Measurement device independent-QKD

Measurement device independent- (MDI-) QKD protocols [161, 162] can be seen as a 'time-reversed' `BB92` protocol: instead of a third party distributing entangled states between Alice and Bob, the two parties send `BB84`-encoded states to a third party, usually referred to as *Charlie*. Charlie subsequently measures the two incoming quantum signals together in the Bell basis, and announces the measurement outcome.

For the correct Bell state outcome (that is obtained with non-unit probability), Alice and Bob are guaranteed to have correlated input states if they used the same basis to encode their states. As such, they do not have to announce the states that they encoded, but merely the bases. This means that Charlie does not need to be trusted, and at the same time that neither Alice nor Bob need to have access to a measurement device, that can be costly and impractical.

Security assumptions

In every QKD protocol there are certain *assumptions* made - sometimes explicit, sometimes implicit. Great care needs to be taken to charter these assumptions, because a protocol that fails to meet these assumption may be rendered insecure. A good example is given by the PNS attack: the assumption that the quantum signal is a single qubit is not met, while security implicitly assumed this to be the case. Many other so-called *side-channel attacks*, colloquially known as *quantum hacking* [17], are made possible by not carefully laying out the assumptions. In a way, one can view DI-QKD as the result of removing as many assumptions as possible.

The *PLOB* bound

The different types of protocols that have been discussed can all obtain different keyrates, where certain types perform notably better than others. Nevertheless, there exists a fundamental limit to key rates that any type of protocol cannot exceed. This bound is known as the *PLOB-bound* and is named after the four authors that introduced it in [163]. It does not assume any structure for a QKD protocol, but instead is derived purely from quantum information theoretic arguments by invoking *channel capacities* [36]. It provides an ultimate limit on asymptotic key rates that any type of DV- or CV-QKD protocol may obtain. However, the bound is for *repeater-less* communication: they can be overcome by introducing *quantum repeaters* [164], which are essential building blocks of a global quantum internet that extend the reach of entanglement by performing *entanglement swapping* (see **Tab.** 2.3).

7.2.3 | Quantitative intuition for security in QKD

In a basic entanglement-based QKD protocol, Alice starts by preparing a Bell pair $|B_{00}\rangle$, after which she sends half of the pair to Bob[6]. Alice and Bob both measure their respective qubits randomly in either the computational or the Hadamard basis. Whenever they picked the *same* basis, their outcomes are perfectly correlated, as writing $|B_{00}\rangle$ in either the Z or X basis shows:

$$|B_{00}\rangle \propto |00\rangle + |11\rangle = |{+}{+}\rangle + |{-}{-}\rangle\,. \tag{7.2}$$

Thus, Alice and Bob perform their measurements in the random bases, and only after obtaining their measurement outcomes they announce their chosen basis. If the bases coincide, they can use their measurement outcomes as the key, since their outcomes are perfectly correlated. If they had opposite bases, they discard their measurement (i.e. *sifting*). By repeating these steps they can create secret key of any length.

The fact that perfect correlations in *two different* bases can be obtained has no classical analogue, and provides the fundament for security of QKD. If the adversary Eve were to intercept the quantum signal, she could e.g. measure the qubit, and then send it to Bob. However, such a measurement will make the qubit collapse to the measurement outcome, which means that Alice's qubit collapses to the same outcome (see (7.2) and the discussion around **Tab.** 2.2). Importantly, Eve has to make a choice whether to measure in the Z or X basis. Consider, for example, that she measures in the X basis, and obtains the outcome $|+\rangle$. After she forwards the qubit, the state that Alice and Bob have is thus $|{+}{+}\rangle$; if Alice and Bob happen to both measure in the Z-basis, their outcomes are *individually* random - they are completely uncorrelated. The possible outcomes are, all with equal probability, $(0,0)$, $(0,1)$, $(1,0)$, $(1,1)$: half of these outcomes are not correctly correlated. Thus, if Alice and Bob verify their outcomes, they find out - with 50% probability - that their outcomes have the wrong parity. A total of m repetitions of these tests would fail to catch Eve with probability 2^{-m}, which is exponentially small in the number of such tests.

Therefore, it is vital that Alice and Bob use part of their measurement outcomes to verify the parities. Because Eve may act maliciously only during those rounds that are not used for verification, it is important that the selection of verification rounds is random, and only selected *after* the measurements have taken place. Alternatively, Alice and Bob could secretly coordinate this choice beforehand, but Eve should not learn this selection.

Moreover, even in the absence of any adversary the correlations will never be perfect due to noise. Any modern protocol allows for some noise, and therefore for some of these verification rounds to fail. Although somewhat

[6] Alternatively, this could be in the other direction, or there could be a server distributing the state. The important point is that, after the communication has happened, Alice and Bob share an EPR pair.

imprecisely stated, the rate of failure of these verification rounds can be referred to as the *error rate* of the implementation. Sometimes, this is called the (X-basis) 'QBER' (*Quantum Bit Error Rate*) or Q_X. Additionally, an estimate must be made for the error rate in the rounds which are not used to verify the signal, but for the actual key generation. This key-generation error rate might be independent from Q_X, and is often named[7] Q_Z or Z-basis QBER. An estimate of both these error rates needs to obtained, which is done during the parameter estimation step[8] of the protocol. Often there is a pre-determined maximum threshold that Alice and Bob have agreed upon, so that when they find a QBER that exceeds this, they abort the protocol.

The attack by Eve proposed above covers only one specific strategy that she might use, so that the analysis so far is not a complete security proof. Taking a more abstract but completer perspective, the security of QKD follows from the *monogamy of entanglement* [165]. Loosely stated, it means that when two qubits A and B are entangled with each other, neither can be entangled with another quantum system C. If $E_{\text{ent}}(A:B)$ is a suitable entanglement measure [35] (like e.g. the entanglement entropy, see Def. 6), this monogamy can be quantified:

$$E_{\text{ent}}(A:B) + E_{\text{ent}}(A:C) \leqslant E_{\text{ent}}(A:BC) \tag{7.3}$$

One consequence from (7.3) is that, if $E_{\text{ent}}(A:B)$ is *maximal*, then

$$E_{\text{ent}}(A:C) \leqslant E_{\text{ent}}(A:BC) - E_{\text{ent}}(A:B) = 0. \tag{7.4}$$

An entanglement measure vanishes on separable states, so the monogamy of entanglement guarantees that if A and B are maximally entangled, E is completely separable from A (and, by extension, from B). In other words, maximally entangled states cannot be entangled with any other system, which mean that they can not be correlated either.

7.3 | Security of QKD

The discussion in sec. 7.2 provides context and intuition for the security of QKD, but it makes no rigorous statements. Moreover, it does not even properly define what it means for a key to be *secure*, let alone how any key generated by QKD adheres to such a definition. This section makes all these topics more precise. First, in sec. 7.3.1 a rigorous definition of security is

[7] In principle 'X' and 'Z' are arbitrary labels for the testing and key generation rounds, but it is often the case that these rounds indeed involve measurements in those specific bases, e.g. in the protocols introduced in chapters 8 and 9.

[8] Sometimes a pre-determined estimate of Q_Z is used instead of estimating it during the protocol run, as it does not affect security to do so, and errors in the key are either solved by error correction, or the protocol is aborted. However, it is vital that Q_X *is* determined during the protocol, as security is derived from it.

given. The methods that obtain security in QKD under this definition are addressed in the rest of the section. Specifically, sec. 7.3.2 discusses *error correction*, the tool to assure that the keys of Alice and Bob are identical. Section 7.3.3 discusses the concept of *privacy amplification*, which is the tool to assure that the generated key is uncorrelated with the adversary Eve, and therefore secret. How these tools exactly provide security under the rigorous definition is detailed by a *security proof* of a protocol, which is discussed in sec. 7.3.4.

7.3.1 | Security definition

Beyond the intuition that it provides, the monogamy of entanglement, as introduced in the previous section, allows for a more quantitative approach. Let A and B be the quantum systems of Alice and Bob, and let E be the combination of all the quantum systems and classical registers that Eve has access to, that together contain her side information. The complete state of A, B and E is ρ_{ABE}, some statistical mixture of pure states in $\mathcal{H}_{ABE}$. If the reduced state $\rho_{AB} = \mathrm{tr}_E\left[\rho_{ABE}\right]$ of Alice and Bob is equal to the Bell pair:

$$\mathrm{tr}_E\left[\rho_{ABE}\right] = |B_{00}\rangle\langle B_{00}|\,, \tag{7.5}$$

then it follows from the monogamy of entanglement that the state ρ_{ABE} must be separable over the bi-partition $AB : E$, so that the full state may be written as:

$$\rho_{ABE} = |B_{00}\rangle\langle B_{00}| \otimes \rho_E, \tag{7.6}$$

for some arbitrary state ρ_E. This is the basis of DI-QKD: if Alice and Bob can confirm that they share the state $|B_{00}\rangle$ (or any other maximally entangled state), they are guaranteed that the state is correlated with *nothing* else, not even other parts of their own hardware, or any secret side channels that Eve might have implemented in either of their labs.

Still, verifying that the entanglement is indeed maximal is not easy; in general, extra steps need to be taken which will affect the key rates. Moreover, (7.6) provides, in a sense, *too much*: what is needed is not a statement on the state that Alice and Bob share w.r.t. the adversary, but rather a statement on how their generated *keys*, k_A and k_B, are correlated with Eve's side information.

Historically, security was defined [166, 167] in terms of the *mutual information* $I_M(k_A : k_E)$ [7, 36] between Alice's key k_A and a (hypothetical) key k_E in the possession of Eve, or the definition was adapted to use the *accessible information* [7, 36] instead. However, intricate problems with this definition mean that the key can only be regarded secure as long as it is *not used in any subsequent application* [168, 169], and thus the definition does not imply any practical security [170].

The problems are solved by the concept of *composable security*, developed originally in the scope of classical cryptography [171] and subsequently adap-

ted to the quantum setting [168]. Notably, the framework of *abstract cryptography* [172, 173] provides an approach where security is defined in terms of closeness to some *ideal* scenario. This scenario is represented by the ideal result of a QKD protocol, the state ρ_{ideal}:

$$\rho_{\text{ideal}} = \frac{1}{|\mathcal{K}|} \sum_{k_A \in \mathcal{K}} |k_A\rangle\langle k_A| \otimes \rho_E, \tag{7.7}$$

where $\mathcal{K}$ is the set of all allowed keys, which is usually the set of all bit strings of a fixed length ℓ. Note that this state is not on the compound system AE, but rather on the compound system $X_A E$, where X_A is a classical register that holds the *key* of Alice. Therefore, the ideal state disregards the quantum system of Alice: this allows, as noted earlier, to derive a statement on the correlation between Eve's side information and just the key of Alice, instead of her entire quantum system. Ultimately, this means that the security of a QKD protocol can be stated in terms of the output key only, and not in terms of the quantum system of Alice (which would, as noted earlier, be doing 'too much').

Note that the ideal state does not contain Bob's key k_B. This is because the definition of security is split into two parts, so that *correctness* and *secrecy* are covered separately. Correctness ensures that k_A and k_B are identical, and secrecy ensures that Alice's key is uncorrelated with any side information of Eve; by extension Bob's key is then secret as well.

No implementation of any QKD protocol will ever be perfect, and therefore only *approximate security* can be obtained. This introduces the need of *security parameters*, usually denoted with $\varepsilon << 1$, that represent the level of security that the protocol provides. The actual state $\rho_{X_A X_B E}$ is then a state with classical registers X_A and X_B holding the keys k_A and k_B of Alice and Bob, and a quantum register E representing all side information in possession by Eve. This state needs to be close to the ideal state; how close is encoded by the security parameter. This allows ε_c-*correctness* and ε_s-*secrecy* to be defined:

Definition 29. [78] *A QKD protocol that outputs k_A and k_B is* ε_c-correct *if:*

$$\Pr[k_A \neq k_B] \leqslant \varepsilon_c. \tag{7.8}$$

Definition 30. [78] *A QKD protocol is* ε_s-secret *if:*

$$D_{\text{tr}}(\rho_{X_A E}, \rho_{\text{ideal}}) \leqslant \varepsilon_s, \tag{7.9}$$

where $D_{\text{tr}}(a, b)$ denotes the trace distance (see (1.29)).

Note that, as mentioned before, Bob's register is dropped in the definition of secrecy. However, this has no effect on security [174]. Finally, a QKD protocol is $(\varepsilon_c + \varepsilon_s)$-secure if it is ε_c-correct and ε_s-secret[9].

[9] In both Defs. 29 and 30 the concept of *robustness* [175] has been omitted. A more

7.3.2 | Error correction

Because the definition of security separates correctness and secrecy, they can be addressed separately. Correctness is addressed first using error correction, which is also known as *information reconciliation* [176]. At this point in the protocol, Alice and Bob are assumed to have keys k_A and k_B that are not necessarily equal, but are at least highly correlated (i.e. Bob's key has some, but not too many errors compared to Alice's key). The *relative* number of errors, i.e. the error rate, can be quantified by e.g. upper bounding the conditional Shannon entropy $H(k_A|k_B)$ (see Def. 3).

To correct these errors, Alice and Bob use an error correction scheme: Alice calculates the *error syndrome* e_s from her key k_A, and sends this over a public channel to Bob. The error syndrome is a bit string that characterizes the key k_A but is considerably shorter than it, and therefore cannot contain a complete characterization of the key. However, the error correction scheme allows Bob, given k_B and e_s, to *decode* a key k'_B that is the same as k_A with extremely high probability:

$$k'_B = \mathrm{dec}(k_B, e_s). \tag{7.10}$$

The length of the error syndrome plays an important role and depends on the chosen error correction code. If e_s is chosen too short, the code may fail to correct all the errors of k_B w.r.t. k_A. At the same time, it should be chosen as short as possible: it must be communicated publicly, so that all information of k_A that is encoded in the syndrome is learned by Eve. The theoretical minimum of the length of e_s is given by the aforementioned conditional entropy [177], but real error correction will suffer from some *inefficiency* $f > 1$. This results in a lower bound of the relative length of the error syndrome:

$$\frac{e_s}{|k_A|} \geqslant f H(k_A|k_B), \tag{7.11}$$

where $|k_A|$ is the length of the key k_A. Still, codes exist that can obtain $f \to_+ 1$ and thus can get arbitrarily close to the theoretical minimum rate.

The quantity $H(k_A|k_B)$ must be estimated, which Alice and Bob do during parameter estimation by e.g. cross referencing part of their keys k_A and k_B. Note that this reduces the effective length of k_A and k_B: this cross-referencing is done using the public channel, so that these communicated bits can not be used as raw key any more.

In the asymptotic limit, (7.11) will reduce[10] to the binary entropy $h_2(Q_Z)$

complete definition would include this, but it is not strictly necessary for the current discussion. Robustness roughly encompasses the notion that a QKD protocol should, in the presence of 'not-too-much-noise', still succeed with a decent probability.

[10] This reduction only works well if the different errors in the bit string are independent from each other, i.e. *uncorrelated* with other errors in the bit string. In the standard practice of DV-QKD, where the measurement outcomes that lead to the raw key are binary

(see (1.40)) of the Q_Z error rate [77]:

$$\frac{e_s}{|k_A|} = h_2(Q_Z). \tag{7.12}$$

Although this is a theoretical limit, there exist many practical codes that can approach the limit for many different error rates Q_Z and key lengths $|k_A|$. A choice that is often used is a *low-density parity check code* (LDPC) [178], for instance that of the `DVB-S2`-standard [179]. Finally, note that instead of estimating Q_Z during the protocol, Alice and Bob can use a pre-determined characterisation of the typical Z-basis QBER.

To obtain an actual quantitative bound on the correctness of the key, Alice and Bob need to verify the error correction scheme. For this they use a *two-universal hashing function* [180, 181], which is a function that takes the key as input and outputs a *hash*: a bit string of length $t < n$. Alice and Bob compute their respective hashes, t_A and t_B, both with length t. Alice communicates her hash t_A to Bob over the public channel, and Bob verifies that $t_A = t_B$. The relevant property of two-universal hashing functions is that, if $t_A = t_B$, it is *extremely* unlikely that $k_A \neq k_B$ [1]:

$$\Pr\left[k_A \neq k_B | t_A = t_B\right] \leqslant 2^{-t}. \tag{7.13}$$

From (7.13) and Def. 29 it is thus immediate that ε_c-correctness is obtained when $t = \log \frac{1}{\varepsilon_c}$. Note that ε_c scales (inverse) exponentially with the hash length, which is a highly desirable property. This means that the hash is relatively small, and independent of the key size.

7.3.3 | Privacy amplification

Privacy amplification, the final step of a QKD protocol, is arguably the most important, because it provides the actual secret key. As stated before, an upper bound to the amount of side information that Eve has can be directly computed from the X-basis QBER Q_X. Additionally, the public communication during error correction (i.e. both the error syndrome and the hash) needs to be accounted for.

Privacy amplification can intuitively be understood as removing any left-over correlation between the key and any side information in the possession of Eve. This step is made possible by once again applying a two-universal hashing function, because of another property of two-universal hashing functions that can be interpreted as the inverse of (7.13): for two inputs to the hashing function that are not exactly the same, it is extremely unlikely that the outputs *are* the same. As such, the outputs are truly uncorrelated, even

outcomes, this is a sound assumption. In CV-QKD the measurement outcomes are continuous, so they are first discretised and then mapped to a bit string before error correction. This makes the (bit-)errors highly correlated between each other, which complicates the error correction process. This is one of the reasons why CV-QKD, although better in theory, performs worse in practice.

though the inputs might be somewhat correlated. Again, the length ℓ of the output of the two-universal hashing function is necessarily shorter than the length $|k_A|$ of the input.

To obtain a truly random key, the key length must be reduced by at least the amount of information that Eve can have about the original key. This is upper bounded by her side information, estimated by Q_X, plus that information she can learn from the public communication. It is hard to exactly determine how much information Eve can learn from this communication, but it can never be more than the lengths (in bits) of the error syndrome and hash.

In the asymptotic limit, the amount of Eve's side information is given by the binary entropy of Q_X, and sec. 7.3.2 explained that the length of the error syndrome is given by the binary entropy of Q_Z. Subtracting these, this results in an asymptotic key rate r_a:

$$r_a = 1 - h_2(Q_X) - h_2(Q_Z). \tag{7.14}$$

Note that (7.14) does not depend on the hash length. The length of the hash is independent of the key length, as explained in sec. 7.3.2, and therefore vanishes in the asymptotic limit. In practice, Alice and Bob often choose the (type of) hashing function in advance, so that they use a fixed error rate Q_{tol} instead of the true Q_X in the privacy amplification. They then verify that Q_X is not above this threshold. The benefit in doing so is that then only one of the two parties needs to be able to estimate Q_X, but it comes with the drawback that technically they could obtain a longer secret key by using Q_X. Furthermore, note that the hashing function should be chosen randomly from a *family* of two-universal hashing functions during the protocol - this choice is usually made by Alice and can be communicated publicly to Bob, but choosing it during the protocol, instead of pre-determining it, ensures security[11].

Note that in the asymptotic limit, the dependence of the key rate on the security parameters has vanished, exactly because the guarantees given by (7.13) and the reverse statement for privacy amplification are only dependent on the length of the hash, but not of the input.

In the finite regime, the key rate will indeed be dependent on the security parameters and block size L. Furthermore, using different security proofs, a single QKD protocol might have different keyrates. Security proofs are discussed in more detail in the next section.

7.3.4 | Finite keys and security proofs

A *security proof* is a complete proof that shows that a specific QKD protocol is $(\varepsilon_c + \varepsilon_s)$-secure, usually for security parameters that can be chosen

[11] In this setting, the two-universal hashing function is a *randomness extractor* [182]. For technical reasons [182–184], this extractor needs a *seed*, an (uncorrelated) random bitstring, which makes the random choice from the family. Recently, *seedless extractors* were considered [185]. However, these cannot function with the conditional min entropy [186] as the *extractor promise*, so the left-over hashing lemma can not be used.

arbitrarily small. In doing so, it gives (an upper bound on) ℓ, the amount of secret key that can be obtained - which is usually dependent on the desired values of the security parameters. Security proofs exist for both general DV-QKD protocols [1] and general CV-QKD protocols [157, 158], and additionally for more specific types of protocols like MDI-QKD [187] and DI-QKD [144].

An indispensable tool in these proofs is the *leftover hashing lemma* [1, 78]. It provides an upper bound to the trace distance from Def. 30 in terms of the smooth conditional min-entropy $H_{\min}^{\varepsilon'}(k_A|\rho_E)$ (see (1.45)), and guarantees that by using privacy amplification an ε_s-secret key can be extracted with the length:

$$\ell = H_{\min}^{\varepsilon'}(k_A|\rho_E) + 2 - 2\log\left(\frac{1}{\varepsilon}\right), \tag{7.15}$$

for any $\varepsilon > 0$ and $\varepsilon' > 0$ s.t. $\varepsilon + 2\varepsilon' \leqslant \varepsilon_s$ [46].

Aided by the leftover hashing lemma, proving security reduces to obtaining a bound on the smooth conditional min-entropy $H_{\min}^{\varepsilon'}(k_A|\rho_E)$. There exist many different techniques that can provide such a bound. If Eve is assumed to attack every round of quantum communication independently and identically (the *i.i.d. setting*), the smooth conditional min-entropy reduces [188] to the smooth von Neumann entropy (see (1.48)), which is much easier to estimate.

This i.i.d. setting is somewhat contrived: Eve may very well combine classical side information of different rounds together (known as *collective* attacks), or perform an attack on all the quantum signals combined (known as a *coherent* attack). It is possible to perform a reduction to the i.i.d. case in these scenarios, by e.g. using the asymptotic equipartition theorem [189] based on de Finetti's theorem [36, 78], or using the related post-selection technique [190]. However, these methods generally give very loose bounds, so that they do not perform well in terms of the secret key length.

Another method of bounding the smooth conditional min-entropy is known as *entropy accumulation* [191, 192]. This method arises naturally in DI-QKD settings but doesn't provide particularly strong bounds either. Recently, *generalized entropy accumulation* [193] was proposed to improve these bounds.

The method of bounding the smooth conditional min-entropy that has seen a lot of success in recent years, uses *entropic uncertainty relations* [1, 194, 195], which are reminiscent of the Heisenberg uncertainty principle [196]. Indeed, the raw key k_A follows from Z-basis measurements on A; (hypothetical) X-basis measurements on the same system wouldn't commute with these Z-basis measurements, so that k_A can be expected to obey some uncertainty relation with the outcomes of such X-basis measurements, denoted x_A. This intuition is quantified by relating the smooth conditional min-entropy $H_{\min}^{\varepsilon}(k_A|E)$ of k_A with the smooth conditional max-entropy $H_{\max}^{\varepsilon}(x_A|B)$ of x_A (see (1.47)). Together they obey the following uncertainty relation [1, 46, 194]:

$$H_{\min}^{\varepsilon}(k_A|E) + H_{\max}^{\varepsilon}(x_A|k_B) \geqslant L, \tag{7.16}$$

where L is the length of the raw key[12]. The smooth conditional max-entropy $H_{\max}^{\varepsilon}(x_A|B)$ is solely determined by the registers in possession of Alice and Bob and does not depend on the adversary Eve; it reduces to the binary entropy of the X-basis QBER in the asymptotic limit.

Note that, in principle, $H_{\max}^{\varepsilon}(x_A|B)$ is a quantity regarding the quantum systems associated with the key generation rounds. Those systems are already measured in the Z-basis for key generation, so that $H_{\max}^{\varepsilon}(x_A|B)$ cannot be directly estimated. However, by using statistical methods it can be estimated from the outcomes of the verification rounds instead; such a specific estimate is included in the security proof in chapter G for the protocol presented in chapter 9.

7.4 Generalization to more than two parties

The generalization of QKD to more than two parties is known as *conference key agreement* (*CKA*) [43, 188]. Besides a single Alice, it involves multiple receivers, usually referred to as the *Bobs* B_i. Alice and all the Bobs together form the *participants*. Both DV (discrete variable) [40, 42, 197] and CV (continuous variable) protocols [41, 198, 199] exist, but only DV protocols are considered in this thesis. In these protocols, various multi-partite entangled states are used, most notably the GHZ state (e.g. in [40, 197]) and the $|W_n\rangle$ state (e.g. in [42]):

$$|W_n\rangle = \frac{1}{\sqrt{n}} \left(|100\ldots0\rangle + |010\ldots0\rangle + \cdots + |000\ldots1\rangle\right). \tag{7.17}$$

In chapter 8 GHZ-based CKA protocols are considered. The basic principle of the GHZ state that allows the participants to generate keys is given by the perfect correlations of Z-basis measurement outcomes: as soon as *any* participant measures in the Z-basis, the state collapses to either $|0\ldots0\rangle$ or $|1\ldots1\rangle$, so that any other participant obtains the same (Z-basis) measurement outcome.

Post-processing steps in CKA are largely the same as for bi-partite QKD. An error correction code that corrects errors with a rate Q can correct errors with a lower rate as well. The maximum bi-partite Z-basis QBER of Alice with every individual Bob can thus be taken as the error rate; Alice announces her error syndrome, so that all Bobs can individually correct their keys. Afterwards, Alice announces the hash of her key, which every Bob individually can compare against the hash of their key. Privacy amplification is even more straightforward, as all participants can apply the hash on their own corrected raw key, resulting in the shared secret key.

[12] The equation as presented here has been simplified by omitting the *complementarity* [1]. Complementarity is a value that reflects how well the measurement bases in a QKD protocol 'complement' each other, which is not always perfect in real setups.

Security in the asymptotic regime can follow from the fact that the GHZ state is *verified* [2, 200]: instead of measuring their qubits in the Z basis to generate key, every participant measures their qubit in either the X or Y basis. Due to the stabilizer nature of the GHZ state, the outcomes of these measurements must be perfectly correlated, provided the total number of Y-basis measurements is even. The participants can thus verify by announcing and subsequently inspecting their measurement bases and outcomes. Repetition of such *verification rounds* allows the participants to obtain a bound estimating how close their state $\rho_{A,B_{\{i\}},E}$ is to the desired state: the GHZ state distributed only between the participants, and some arbitrary state ρ_E that is completely separable form the participants' quantum state.

In the finite regime, the verification of the underlying GHZ state could offer security as well, but (similar to the QKD case) the security can also be stated immediately on the generated key itself. Indeed, the leftover hashing lemma (see (7.15)) paired with a suitable selection of estimation method for the smooth min-entropy remove the need of verifying the underlying GHZ state.

Verification and security will be addressed in more detail in chapters 8 and 9. These two chapters present different CKA protocols that are additionally *anonymous*, where beyond the message itself, the *identity* of the participants is hidden from the rest of the network as well.

7.5 | Anonymity in networking protocols

QKD and CKA can provide *security* in communication, so that parties in a network can communicate without anyone else in the network being able to learn the contents of their communication. Nevertheless, by running the protocols, it is clear for anyone that can monitor the network traffic that the participants *are* communicating. It may be the case that e.g. Alice wishes to hide not the contents of her message, but her identity as the origin of the message. That is, she wishes to remain *anonymous*: her identity as the sender remains hidden to anyone else in the network, even after the protocol has ran. Similarly, Bob as the designated receiver may wish to remain anonymous, although in this case it is only sensible to hide his identity from anyone in the network except Alice.

These examples show a more general point: anonymity should be defined *with respect to the other parties in the network*. This allows for different 'levels' of anonymity: e.g. Alice may wish to hide her identity from anyone in the network except her chosen receiver Bob, or additionally even from Bob himself.

In the setting with more than two parties, this can generalize to even more different settings. In a CKA protocol where Alice chooses any number of Bobs from the network as receivers, she may wish to hide her identity from

everyone in the network, or just from those nodes that she didn't choose to be a receiver.

Simultaneously, she may wish that the designated receivers are only aware of their own role, but not who else in the network is a receiver. In contrast, if the receivers know who in the network are the other receivers, a weaker level of anonymity is obtained. In such a setting the participants merely form a special 'group' in the network; the rest of the network is then referred to as the *non-participants*.

Defining anonymity has to take all these considerations into account. Moreover, consider the perspective of any non-participant that may wish to learn the identity of e.g. Alice. For this non-participant, determining which node in the network is Alice can be regarded as a 'guessing game'.

A slightly naive first approach to defining anonymity could be in terms of equal guessing probabilities for every node in the network. This would be problematic in asymmetric networks, however. Indeed, consider a network where one node is much more likely to be a sender than another node, purely based on e.g. its physical location. Here, a definition that treats every node as equally likely to be the sender would not be applicable.

Early definitions [26, 201] do not necessarily consider this point: anonymity is defined in terms of an equal probability for every node to be a sender or receiver. Following Pub. **[A]** ([2]), anonymity can be defined in terms of any *extra* information that the adversary can learn during the protocol. This extra information $\mathcal{I}_+$, which includes all public communication and any quantum systems that the adversary has access to, cannot alter the probability of a node taking a certain role.

Definition 31. (Pub. **[A]**) *Let* $\mathbf{P} \subset \mathcal{N}$ *be the set of participants of an anonymous protocol in a network* $\mathcal{N}$*, and let Eve be an adversary that wishes to learn* $\mathbf{P}$*. Furthermore, let* $\mathcal{I}_{\text{Eve}}$ *be the information regarding* $\mathbf{P}$ *that Eve has both beforehand and trivially learns by corrupting any number of non-participants. The protocol is* anonymous *if, for every subset* $\mathbf{G} \subset \mathcal{N}$*:*

$$\Pr\left(\mathbf{G} = \mathbf{P} | \mathcal{I}^+_{\text{Eve}}, \mathcal{I}_{\text{Eve}}\right) = \Pr\left(\mathbf{G} = \mathbf{P} | \mathcal{I}_{\text{Eve}}\right), \tag{7.18}$$

where $\mathcal{I}^+_{\text{Eve}}$ *is the information that Eve additionally learns during the protocol, which includes all public communication and all quantum systems she has access to.*

Def. 31 ensures that anonymity can remain intact even if Eve corrupts any number of non-participants. Upon learning the information $\mathcal{I}^+_{\text{Eve}}$, that now includes the quantum systems of those corrupted non-participants, the probability distribution is unaffected, so that every subset $\mathbf{G} \subset \mathcal{N}$ is as likely to be the set of participants as without learning $\mathcal{I}^+_{\text{Eve}}$.

However, Def. 31 does not provide a *measure* of anonymity: under its definition, a protocol either is or is not anonymous. On the other hand, *approximate anonymity* - similar to *approximate security* (see Defs. 29 and 30)

- involves an *anonymity parameter* ε_a that ensures that a protocol can be anonymous under a suitable notion, without having to be *perfectly* anonymous. This is provided by defining ε_a-*anonymity*.

Definition 32. (Pub. [C]) *Let $\rho_{\mathcal{P},C,E}$ be the state of any protocol, where $\mathcal{P}$ is a (classical) register that holds the information regarding* **P**, *C is a register that contains all public communication of the protocol, and E is a (quantum) register reflecting all side information of Eve, which includes quantum systems. Furthermore, let $\sigma_{\mathcal{P},C,E}$ be any* ideal *state that is anonymous per Def.* 4 *of* Pub. [C]. *Then, the protocol is* ε_a-anonymous *if*

$$D_{\mathrm{tr}}(\rho_{\mathcal{P},C,E}, \sigma_{\mathcal{P},C,E}) \leqslant \varepsilon_a, \tag{7.19}$$

where $D_{\mathrm{tr}}(\rho, \sigma)$ denotes the trace distance between ρ and σ (see (1.29)) and where this inequality must hold for any choice of participants **P**, *or more specifically for any choice of sender $\mathcal{A}$ and Bobs $\{\mathcal{B}_i\}$.*

Def. 32 as an alternative definition of anonymity provides a notion of approximate anonymity, which could make it composable, although this has not been proven [48]. Furthermore, it allows for easy adaptation towards other notions of anonymity. In particular, the ideal state σ can be replaced by another state that is merely *partially* anonymous (see Def. 3 in [48]): a partially-anonymous protocol provides anonymity of the participants merely from the perspective of the non-participants or Eve, but allows the participants to know each others' identity. This is in contrast to the above level of anonymity (hence referred to as *fully anonymous*), in which only Alice knows the identity of the Bobs; the Bobs are not aware of the selection of **P** beyond their own role.

7.6 | Conclusion

This chapter has introduced all the relevant concepts and definitions regarding QKD and CKA that are used in the subsequent chapters of part III. Chapters 8 and 9 both introduce protocols for *anonymous conference key agreement* (ACKA); the first chapter contains protocols for *star networks*, while the latter chapter contains a protocol for *linear networks* instead. Furthermore, chapter 9 contains a complete finite key analysis of the protocol that is introduced in that chapter, although some of the technical details of the analysis and proofs are deferred to the appendices. Chapter 10, the last chapter of part III, details the experimental realisations of the protocols introduced in both chapters 8 and 9.

8

Anonymous Conference Key Agreement in Star Networks

Chapter 7 has introduced the concept of *Quantum Key Distribution* (QKD) and its generalization *Conference Key Agreement* (CKA) as important goals within quantum networking and communication. Any protocol that performs CKA allows multiple parties in a network to create a secret hidden key between just themselves, while excluding any other party in the network from accessing the shared key. Such *conference key* can subsequently be used is a versatile range of cryptographic tasks, including private communication [17], secret sharing [202–204] and multi-party computation [22, 25].

Additionally, chapter 7 introduced the concept of *anonymity* within the same quantum communication setting. Anonymity is a desirable property that networking protocols can have, so that the *identities* of the involved network parties remain hidden (in addition to the normal intended purpose of the protocol, like e.g. key distribution or secret sharing).

This chapter covers both Pubs. **[A]** and **[C]**. More specifically, it addresses the combination of anonymity and conference key agreement. Such *anonymous conference key agreement* (ACKA) allows a subset of parties in the network to create a shared secret key without revealing their identity to the other parties in the network, or potentially even to each other. The first protocol

that performs ACKA was introduced in Pub. **[A]** ([2]). However, some issues exist with this first protocol, which renders it more a 'proof-of-concept' and impractical for real-world implementation. To address its shortcomings, a second ACKA protocol was introduced in Pub. **[C]** ([48]). In fact, this protocol comes in two different versions, that cater to *partial* and *full* anonymity, respectively (see sec. 7.5). Moreover, Pub. **[C]** additionally contains a complete finite key analysis of the presented protocols, and a more complete security proof. This is not presented in this thesis, as it closely resembles the analysis for a related protocol that will be presented in chapter 9.

The setting for the protocols, including the network topology and security, is made more precise in sec. 8.1. Subsequently, sec. 8.2 contains the first ACKA protocol: in sec. 8.2.1 the protocol is stated and an analysis is given regarding the correctness and security. The anonymity of the protocol is addressed separately in sec. 8.2.2, and the aforementioned issues are discussed in sec. 8.2.3.

Then, sec. 8.3 contains the two versions of the improved protocol. They are both introduced in sec. 8.3.1; their analysis regarding correctness, security and anonymity is addressed in sec. 8.3.2. How the improvements over the original protocol affect the performance is discussed in sec. 8.3.3, which additionally presents the finite key rates. The fully anonymous version of the protocol makes use of an adapted definition of anonymity, which is described in sec. 8.3.4. Pub. **[C]** additionally contains an assessment of the performance of the protocols in a real-world scenario by simulating it and comparing it against an ACKA protocol that does not involve multi-partite entangled states. This is not presented in detail in this thesis, but only addressed briefly in sec. 8.4. In that same section, a brief discussion of the shared network topology of all different protocols can be found, as well as a conclusion to the chapter.

8.1 Setting for the security and the protocols

It is helpful for the introduction and subsequent analysis of the protocols to define the setting, and the network on which the protocol is run. The protocols that this chapter covers all have a *central server* that distributes a quantum state over the entire network. This server does not take part in the protocol as a node, and is therefore viewed separately form the network, which is denoted with $\mathcal{N}$. The network is referred to as a *star* network, because the topology is such that all nodes are connected to the central server, while no quantum connections between the nodes are implied. Depending on the specific protocol, there might be various levels of trust imposed on the central server; this is addressed in more detail in secs. 8.2 and 8.3 and especially in sec. 8.4. In the spirit of the network being a star network, all protocols presented in this chapter involve a $|\mathrm{GHZ}_{\mathcal{N}}\rangle$ state as a resource, distributed by the central server.

Partitioning of the network

The network $\mathcal{N}$ can be divided into different sets that reflect the identities or roles of the different nodes. More specifically, the network consists of $n = |\mathcal{N}|$ nodes and includes a special node $\mathcal{A} \in \mathcal{N}$ called *Alice*; Alice wishes to perform CKA. She picks $m \leqslant n - 1$ nodes, referred to as the *Bobs* $\mathcal{B}_i$, with whom she wishes to establish a secret key. The set of Alice and all Bobs together is referred to as the *participants* $\mathbf{P}$, and the rest of the network is referred to as the *non-participants* $\bar{\mathbf{P}} = \mathcal{N} \setminus \mathbf{P}$, which is prohibited from learning the key.

Additionally, Alice wishes for both her and the Bobs to remain *fully anonymous*, so that any node in the network learns nothing about the role of any node in the network besides themselves (with the exception, of course, of $\mathcal{A}$, because she chooses the set of participants). Alternatively, one version of the protocol in sec. 8.3 is *partially anonymous*, where the Bobs are aware of the set of participants as well, including the special identity of $\mathcal{A}$.

The non-participants may be *honest-but-curious*, which means that they will follow the steps that the protocol prescribes for them, even though they are still interested in learning the identity of either Alice or the Bobs. Alternatively, any non-participant can be *corrupted*, which means that they can act maliciously to find out either the secret key or the identity of anyone. This includes actively deviating from the protocol and *colluding* with any other corrupted non-participant; they are represented by an adversary *Eve*. Eve could, in principle, be not part of the network, but no generality is lost in assuming that she is in $\mathcal{N}$. The collection of all honest-but-curious participants is denoted $\mathbf{H}$, and the collection of all corrupted non-participants is denoted $\mathbf{C}$, so that $\bar{\mathbf{P}} = \mathbf{H} \cup \mathbf{C}$. **Fig.** 8.1 presents an overview of the network partitioning.

Note that it is assumed that no participant is corrupted: this would defeat the purpose of the scheme, as any other corrupted, colluding party would then have access to the secret key. However, the participants are still *honest-but-curious*: they are interested in learning the identity of Alice or the other Bobs, even though they follow the protocol. In this sense the anonymity of the protocol needs to be addressed from two different perspectives: from the perspective of the entire set $\mathbf{C}$ (which includes Eve), and from the perspective of a single Bob in $\mathbf{P}$. The case of a single node in $\mathbf{H}$ is then implicitly covered by the $\mathbf{C}$ setting.

Note that all participants and non-participants have access to all public communication throughout the protocol, so even the honest-but-curious nodes may use this to infer the partitioning.

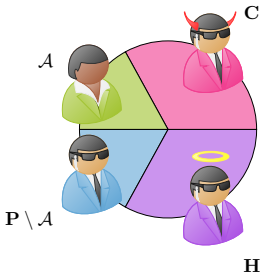


Figure 8.1: The entire network $\mathcal{N}$ can be partitioned into four disjoint subsets of nodes. Alice, denoted $\mathcal{A}$, takes a special role in the protocol; she chooses a set of m Bobs $\mathcal{B}_i$ to share a key with. Together, Alice and the Bobs form the *participants* $\mathbf{P}$. The rest of the network are the *non-participants* $\bar{\mathbf{P}}$; they can be further divided into the *honest-but-curious* nodes $\mathbf{H}$, who follow the steps of the protocol but may still wish to learn the identity of $\mathbf{P}$, and the *corrupted* nodes $\mathbf{C}$, who act maliciously by actively deviating from the protocol and colluding with each other. The protocol is *anonymous*: only Alice is aware of the partitioning of the network (except how $\bar{\mathbf{P}}$ is divided into $\mathbf{H}$ and $\mathbf{C}$). $\mathbf{P}$ is notified of their role during the protocol. In the *fully* or *partially* anonymous setting (see sec. 7.5), the participants do not or do know the identity of $\mathcal{A}$ and the other participants, respectively. $\bar{\mathbf{P}}$ is not aware of the identities of $\mathbf{P}$.

8.2 | Original protocol

The original protocol, `ANONYMOUS CONFERENCE KEY AGREEMENT` (`ACKA`), consists of various steps that include three different sub-protocols; these are presented in sec. 8.2.1, where also the main protocol is analysed. Anonymity is addressed in sec. 8.2.2. The protocols' performance is addressed in sec. 8.2.3.

To ease notation in the analysis, the participants are (whenever applicable) identified with indices i, whereas the non-participants are identified with indices j. Moreover, w.l.o.g. the participants are assumed to be the first $m+1$ nodes in the network, so that their qubits can be referred to as nodes 0 (for Alice) and nodes $1, \ldots, m$ for the Bobs. This means that the non-participants are the nodes $\{j\}_{m+1 \leqslant j \leqslant n-1}$, for a total of n nodes.

8.2.1 | Protocol statement and analysis

The protocol makes use of of three subprotocols. These are named and explained briefly here; a full statement and analysis can be found in chapter D.

1. `NOTIFICATION` allows $\mathcal{A}$ to anonymously notify every other participant $\mathcal{B}_i$ that they are a participant. It is a purely classical protocol originally presented in [205], but requires private channels between every pair of nodes in the network. The protocol is introduced in more detail in sec. D.1.

2. `ANONYMOUS MULTIPARTITE ENTANGLEMENT` (`AME`) allows the participants $\mathbf{P}$ to anonymously extract a $|\mathrm{GHZ}_{\mathbf{P}}\rangle$ state on just their qubits from the $|\mathrm{GHZ}_{\mathcal{N}}\rangle$ state. During this protocol, the non-participants $\bar{\mathbf{P}}$ measure their qubit in the X basis and announce the outcomes x_j. Based on these outcomes, $\mathcal{A}$ performs a correction on the GHZ state. See **Fig.** 8.2 for a visualization. The protocol is introduced in more detail in sec. D.2.

3. `VERIFICATION` allows $\mathcal{A}$ to verify that the state after `AME` is indeed the expected state, implicitly verifying the behaviour of $\mathbf{P}$, $\mathbf{H}$ and especially $\mathbf{C}$. During this protocol, the participants $\mathbf{P}$ measure their qubit either in the X or Y basis, encoded by a random bit b_i; they announce these with the outcomes as (b_i, o_i). $\mathcal{A}$ uses these announcements to verify that the outcomes have the correct parity that the GHZ state should generate. The protocol is introduced in more detail in sec. D.3.

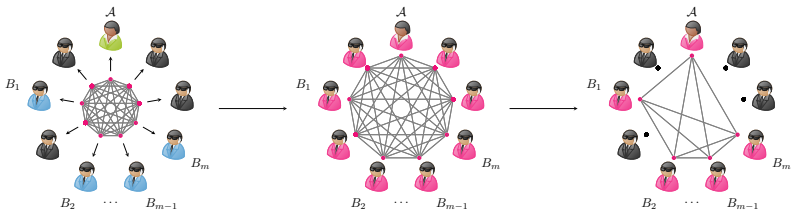


Figure 8.2: Visualization of `AME`. First, a $|\mathrm{GHZ}_{\mathcal{N}}\rangle$ state is distributed by the central server between all nodes of the network. Even though the participants secretly play a special role, their aim is to be indistinguishable from all other nodes in the network. During the protocol, all non-participants $\bar{\mathbf{P}}$ measure their qubit in the X-basis, while the participants $\mathbf{P}$ do nothing; after a correction by Alice the state of the network is $|\mathrm{GHZ}_{m+1}\rangle$ for the participants, disentangled from all other nodes in the network.

Using these subprotocols `ACKA` can be defined, presented as `Protocol I`.

Protocol I - ANONYMOUS CONFERENCE KEY AGREEMENT	
Input:	Alice as initiator; parameters L and D.
Goal:	Anonymous generation of secret key between $\mathbf{P}$.

1: Alice runs NOTIFICATION to notify the m Bobs.

2: The source distributes L $|\mathrm{GHZ}_{\mathcal{N}}\rangle$ states.

3: For each of the L $|\mathrm{GHZ}_{\mathcal{N}}\rangle$ states, the network runs AME to extract a $|\mathrm{GHZ}_{\mathbf{P}}\rangle$ state on the nodes of the participants. $\bar{\mathbf{P}}$ announces their measurement outcomes $\{x_j\}$, $\mathbf{P}$ announces random bits $\{x_i\}$.

4: For each of the L $|\mathrm{GHZ}_{\mathbf{P}}\rangle$ states, the parties ask a public source of randomness to broadcast a bit b such that $\Pr[b = 1] = 1/D$.

Verification: If $b = 0$, $\mathbf{P}$ runs VERIFICATION on the $(m+1)$-partite state, announcing the measurement basis and outcome (b_i, o_i). $\bar{\mathbf{P}}$ announces random pairs of bits (b_j, o_j).

Keygen: If $b = 1$, $\mathbf{P}$ measures in the Z basis to obtain a key bit.

5: If Alice accepts all VERIFICATION rounds, she anonymously validates the protocol.

A flowchart detailing the basic steps of the protocol can be found in **Fig.** 8.3. First, in step 1, Alice uses the NOTIFICATION protocol to ensure the Bobs are aware of their role, while maintaining her anonymity. Next, in step 2, a fixed number L $|\mathrm{GHZ}_{\mathcal{N}}\rangle$ states are distributed over the entire network. L is pre-determined and referred to as the *block size*.

From each of these $|\mathrm{GHZ}_{\mathcal{N}}\rangle$ states, a $|\mathrm{GHZ}_{\mathbf{P}}\rangle$ state on only the participants $\mathbf{P}$ is extracted by running AME during step 3. In this protocol, the non-participants $\bar{\mathbf{P}}$ perform a measurement on their qubit so that they are removed from the $|\mathrm{GHZ}_{\mathcal{N}}\rangle$ state. To obtain the $|\mathrm{GHZ}_{\mathbf{P}}\rangle$ state, $\mathcal{A}$ has to perform a correction to the network state which is based on the measurement outcomes $\{x_j\}$ of $\bar{\mathbf{P}}$. Indeed, if the parity of all these measurement outcomes is odd, the state would have an incorrect phase $|0\ldots0\rangle_{\mathbf{P}} - |1\ldots1\rangle_{\mathbf{P}}$; $\mathcal{A}$ can correct this by applying a $Z_{\mathcal{A}}$ operator to her qubit.

Hence, the non-participants $\bar{\mathbf{P}}$ have to communicate their outcomes to $\mathcal{A}$, which they do by announcing them publicly. To hide their role, the participants announce random bits $\{x_i\}$.

Subsequently, in step 4 this state is either verified using the VERIFICATION protocol (a **Verification round**), or used to generate a bit of raw key (a **Keygen round**). The verification rounds ensure that the used states are ε-close to the $|\mathrm{GHZ}_{\mathbf{P}}\rangle$ state, where ε is exponentially small in the number of

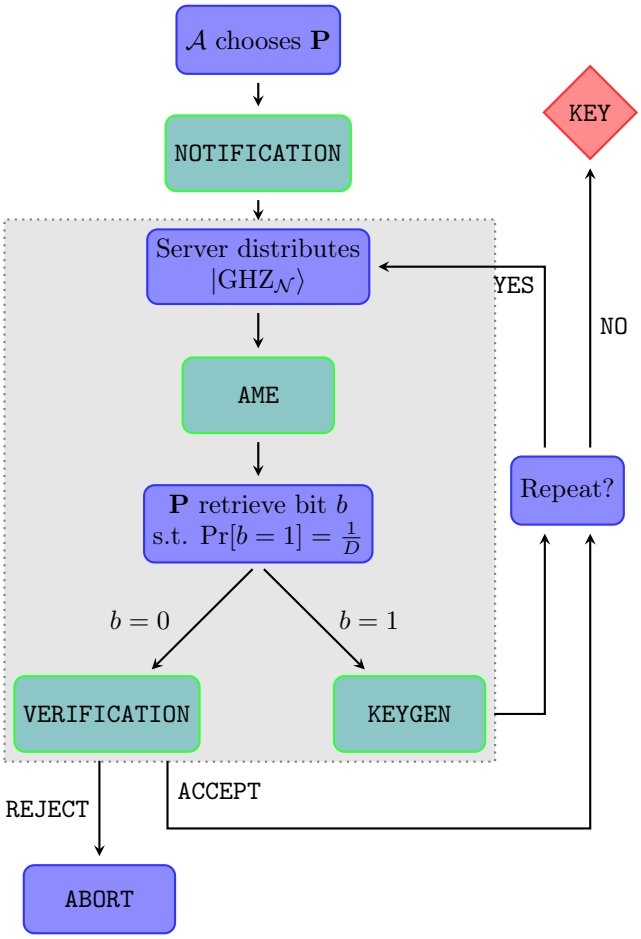


Figure 8.3: A flowchart detailing the steps and subprotocols of ACKA; the green boxes are the subprotocols which are detailed in chapter D. The gray box is repeated until enough key has been created, after which the key is outputted.

Verification rounds. There are $L(1-\frac{1}{D})$ such rounds, so the asymptotic key rate of the protocol is $\frac{L}{D}$.

Analysis

The NOTIFICATION subprotocol allows Alice to anonymously communicate a bit to everyone in the network separately, to indicate if they are a participant or not. Instead of using the complete ACKA protocol to anonymously establish

a secret key, she could use NOTIFICATION instead to communicate a random bit to only those nodes in **P**. This would effectively anonymously establish a secret key with the desired parties, bypassing the need for the rest of ACKA. However, NOTIFICATION has $\mathcal{O}(n)$ rounds for one bit of key, and every round needs pairwise private communication, for a total of $\mathcal{O}(n^3)$ necessary private channel uses per generated bit. Implementing this consumes a lot of private key bits, so ACKA can be viewed as an improvement on this classical scheme with better scaling properties.

As presented in sec. D.3, VERIFICATION is a protocol that runs only on the nodes in **P**. The protocol involves communication, which would immediately break anonymity: every node in the network that performs an announcement, is then automatically a participant. Similar to AME, the nodes in $\bar{\mathbf{P}}$ therefore announce random pairs of bits to hide their identity[1]. Even though the announcements are indistinguishable for anyone else in the network, Alice (having chosen **P**) can determine what are the 'true' measurement bases and outcomes, so that she can perform the verification. It remains to be proven that all these announcements are indeed indistinguishable for anyone else, so that the participants remain anonymous; this is addressed in sec. 8.2.2.

Technically, the **Verification** rounds only verify the states that are used for verification, but never the GHZ states from the rounds that are used for key generation. It is therefore of vital importance to postpone the choice of round type until *after* the state has been distributed: otherwise, the adversary, potentially having access to the source, could 'play nice' during the **Verification** rounds by distributing the correct states, and then distribute arbitrary different states during the **Keygen** rounds. In particular, the adversary could distribute an $(n+1)$-qubit GHZ state, secretly keeping one qubit for themselves, and perform a Z-basis measurement - thereby learning the key and completely breaking security.

In the same spirit, the choice of round type has to be performed *after* AME is used to extract the GHZ state on the participants. In particular, until after all non-participants have announced their random bits - this ensures that no non-participant can freely choose to adhere to the protocol during only those rounds where their behaviour is checked. Moreover, the public source of randomness must be trusted, in the sense that no adversary can choose or determine beforehand the bit b. At the same time it is no issue that any non-participant or adversary learns the value of b when the participants learn it - they have already committed to all their communication, so can not alter their strategy based on the value of b any more.

The protocol dictates that all L GHZ states are distributed at the same time. This implies that all nodes must have access to a large quantum memory to store their qubits. However, in practice this can be performed in repeated

[1] Of course, any dishonest party in **C** may not make this announcement. However, the only effect this has is that they effectively announce that they are not a participant, which they can already do anyway.

steps (i.e. the gray box in **Fig. 8.3**) that can be performed one-by-one, merely storing the raw key from the **Keygen** rounds.

Moreover, the presentation of the protocol is very modular, as the subprotocols are all self-contained. However, certain steps can be skipped when the protocol is seen as a whole. As an example, the correction that $\mathcal{A}$ performs during the `AME` step is strictly speaking not necessary - she would only have to verify for a slightly different state during `VERIFICATION`, which results in that she only `ACCEPT` when the opposite parity is found.

Finally, another useful property of the protocol design is that by performing the `VERIFICATION` steps, the `NOTIFICATION` step are implicitly verified as well. If the wrong set is notified to be the participants, Alice takes the wrong announcements into account - at least one person that she believes is in $\mathbf{P}$ has in fact announced a completely random (b_j, o_j) pair, or she does not take into account all true announcements. In these cases, the announcements that Alice takes into account do not possess the correct correlations, so that verification will fail.

8.2.2 | Anonymity

The anonymity of `ACKA` follows intuitively from the intrinsic correlations of the GHZ state and its non-local phase. By applying a Z operation, any node of a GHZ state can induce a non-local effect on the state:

$$Z_i \left|\mathrm{GHZ}_n\right\rangle = \frac{1}{\sqrt{2}} \left(\left|00\ldots0\right\rangle - \left|11\ldots1\right\rangle\right). \tag{8.1}$$

Since this state is independent of the node i to which the Z operation was applied, the non-local affect is indeed obtained. The effect of the reverse operation is non-local as well, and therefore the correction by $\mathcal{A}$ based on the announced measurement outcomes in `AME` does not disclose her identity.

	AME	Verification
$\mathcal{A}$	random bit x_0	random bits (b_0, o_0)
$B_i \in \mathbf{P}\setminus\mathcal{A}$	random bit x_i	random bit b_i, outcome bit o_i
$P_j \in \mathbf{H}$	outcome bit x_j	random bits (b_j, o_j)
$P_k \in \mathbf{C}$	arbitrary bit $\tilde{x}_k$	arbitrary bits $(\tilde{b}_k, \tilde{o}_k)$

Figure 8.4: Overview of the public communication throughout `ACKA`. During `AME` and `VERIFICATION`, the different subsets in the network announce either measurement outcomes or random bits; these all need to be indistinguishable to prevent anyone from learning the identities of the involved parties.

During the various steps of the protocol, several measurement outcomes and bases must be communicated to $\mathcal{A}$ (see **Fig. 8.4**). Since no one in the

network is aware of the identity of $\mathcal{A}$, these outcomes and bases are communicated by a broadcast, so that they become completely public:

- The outcomes $\{x_j\}$ of $\bar{\mathbf{P}}$ during `AME`, allowing $\mathcal{A}$ to correct the state.
- The bases $\{b_i\}$ of $\mathbf{P}$ during `VERIFICATION`, allowing $\mathcal{A}$ to verify the state.
- The outcomes $\{o_i\}$ of $\mathbf{P}$ during `VERIFICATION`, allowing $\mathcal{A}$ to verify the state.

As **Fig.** 8.4 shows, the other parties (i.e. $\mathbf{P}$ and $\bar{\mathbf{P}}$ during `AME` and `VERIFICATION`, respectively) announce uniformly random bits to not inadvertently give away anyone's identity. This means that the announcements are inherently different in nature for $\mathbf{P}$ and $\bar{\mathbf{P}}$, so that their difference might be exploited by anyone to determine the identities of the participants. To guarantee anonymity, all the announcements must be indistinguishable from the uniformly random bits. Since the choice of basis b_i is an individual, uniformly random choice for every node in $\mathbf{P}$, it is straightforward that they are indistinguishable from the announcements of the other nodes.

The measurement outcomes $\{x_i\}_{\bar{\mathbf{P}}}$ and $\{o_i\}_{\mathbf{P}}$ warrant a more careful analysis. Beyond being individually random, there must not be any correlations between the different announcements as well. Note that $\mathcal{A}$ alone, knowing the set $\mathbf{P}$, is able to distinguish the true measurement outcomes from the uniformly random announcements, so that she can perform the necessary correction and verification. A detailed proof of indistinguishability can be found in chapter E.

The **Verification** rounds ensure that the state on $\mathbf{P}$ is ε-close to the $|\mathrm{GHZ}_{\mathbf{P}}\rangle$ state for some small ε, and thus dis-entangled from $\bar{\mathbf{P}}$. Simultaneously, the non-participants in $\mathbf{H}$ measure their qubit during `AME` and thus become disentangled from the network. Thus, at the start of a **Keygen** round the state $|\mathcal{N}_{\mathbf{Keygen}}\rangle$ of the network is:

$$|\mathcal{N}_{\mathbf{Keygen}}\rangle \hat{=} |\mathrm{GHZ}\rangle_{\mathbf{P}} \otimes |\mathbf{H}\rangle \otimes |\Psi\rangle_{\mathbf{C}}\,, \tag{8.2}$$

where $|\mathbf{H}\rangle = \bigotimes_{j\in\mathbf{H}} H_j\,|x_j\rangle_j$ is the post-measurement state of the honest non-participants who obtained the outcomes $\{x_j\}$, and H_j is the Hadamard operator on node j (see **Tab.** 1.1). The state $|\Psi\rangle_{\mathbf{C}}$ is an arbitrary (purification of) the state of the colluding parties $\mathbf{C}$. There is no communication during the **Keygen** rounds, and the state is separable between the sets $\mathbf{P}$, $\mathbf{H}$ and $\mathbf{C}$, so there can be no leakage of identity during the rounds. A more detailed analysis that addresses the anonymity from all different perspectives (a honest-but-curious Bob, a honest-but-curious non-participant in $\mathbf{H}$, or the colluding parties $\mathbf{C}$) can be found in sec. E.3.

8.2.3 | Performance

ANONYMOUS CONFERENCE KEY AGREEMENT, as presented in Pub. **[A]** ([2]), was the first proposal of anonymous conference key agreement, but it suffers from some drawbacks that render it more a proof-of-concept, instead of a robust protocol. These issues include:

1. The key rate of the protocol is low: on average $\frac{1}{D}$ bits of key are created for every network GHZ state. Even with moderate security requirements the parameter D is relatively large, which means that the key rate can never improve beyond orders of magnitude below unity.

2. Related to the previous point, the security of the generated key is derived from a bound on the quantum state, namely that the generated state is ε-close to the GHZ state (on $\mathbf{P}$). This is done by performing many different measurement settings, (i.e. the 2^{n-1} different choices of measurement basis in the VERIFICATION protocol), which is essentially doing too much. Proving security directly on the created key (through the use of the left-over hashing lemma and e.g. entropic uncertainty relations, see sec. 7.3.4) would allow for considerably fewer measurement settings and rounds for security verification. This would result in a much higher key rate.

3. As presented, the protocol is completely non-robust to noise or other imperfections in the implementation. Even if just a single verification round fails (through e.g. a faulty measurement by one of the participants), the protocol aborts.

4. The security analysis derived from the ε-closeness of the state does not cover coherent attacks, but just the i.i.d. setting. To address these shortcomings, tools like de Finetti's theorem (see sec. 7.3.4) need to be used to perform a reduction. However, these steps would decrease the key rate even further.

5. Anonymity is proven (as detailed in chapter E) in terms of Def. 31 instead of Def. 32.

6. Provided the adversary has control over the source, and does not care about the protocol being successful, they can easily break anonymity. Indeed, if the source distributes $+1$ eigenstates of the X operator (i.e. the $|+\rangle$ state), all (honest) non-participants would obtain outcome $x_j = 0$ during AME. The participants have $x_i = 1$ with probability half, so any node announcing 1 during AME necessarily gives up their identity as a participant. VERIFICATION implicitly verifies the distributed state to be $|\mathrm{GHZ}_{\mathcal{N}}\rangle$, as otherwise the correct correlations can not be produced. However, for anonymity this is too late, as VERIFICATION has to happen *after* AME.

These problems, most notably items 1 to 4, make the protocol unsuitable for a real-world implementation, especially with current or modest-future technology. Moreover, although they are mentioned in [2] as steps that should be performed, the protocol does not explicitly include any error correction or privacy amplification steps.

8.3 Improved protocols

To address the problems and shortcomings listed in sec. 8.2.3, a new protocol, `ANONYMOUS CONFERENCE KEY AGREEMENT VERSION 2` (`ACKAv2`), was introduced in Pub. **[C]** ([48]). In fact, two different protocols were introduced, one of which obtains *partial* anonymity, and one of which obtains *full* anonymity (see sec. 7.5). In contrast to `Protocol I`, both these protocols are shown to be anonymous following definition Def. 32. However, the second protocol only obtains full anonymity under an extra assumption on the network - therefore the definition of anonymity is adapted to allow for this assumption. Moreover, the two protocols introduce a new assumption, namely the bounded storage model assumption. It is assumed that the nodes in the network (but **not** the adversary) have a quantum memory with only limited storage time (known as the *bounded storage* model [206]). Moreover, as the protocol is a *key-expanding scheme*, the participants need some pre-shared secret key.

The two protocols are introduced first in sec. 8.3.1, after which differences with `Protocol I` are addressed and explained in sec. 8.3.2, where additionally an explanation is given how the issues listed in sec. 8.2.3 are solved. The key rates of the new protocols are addressed in sec. 8.3.3, and the aforementioned issue in the fully anonymous setting is detailed in sec. 8.3.4, including the explanation of why an adaptation is necessary.

8.3.1 Protocol statement

The improved protocols make use of a selection of subprotocols, that are introduced and detailed in Pub. **[C]** ([48]):

1. `IDENTITY DESIGNATION`: up to a few key differences and additions, this protocol has a goal similar to `NOTIFICATION`. Most notably, beyond notifying the Bobs of their role, it additionally performs *collision detection*, so that there is a guarantee that there is only one sender.

2. `PARITY`: this protocol allows the network to anonymously compute the parity of a set of input bits held by each node. The protocol is used to communicate the *testing key* (introduced in step 2 of `Protocol II`) to the non-participants in step 4 of `Protocol II`. It is additionally used in step 5 of `Protocol II` to communicate the parity of the measurement outcomes of the test rounds. The version used in the main protocol as presented in Pub. **[C]** is adapted from [205], but does not need a simultaneous broadcasting channel.

3. `TESTING KEY DISTRIBUTION`: this subprotocol is only necessary in the fully anonymous protocol. It allows $\mathcal{A}$ to share the testing key to the other participants, without them having to know each other. It is quite costly to implement, considerably affecting the key rate of the fully anonymous protocol.

4. `ANONYMOUS ERROR CORRECTION`: this subprotocol allows the participants to anonymously perform error correction to ensure correctness. A partially anonymous version is used in step 7, and a fully anonymous version is used in the second protocol; both are introduced in Pub. **[C]**. A related but different protocol is introduced in more detail in chapter 9.

Using these subprotocols, `ACKAv2` can be defined, presented as `Protocol II`, to be found on the next page.

Instead of stating the entire fully anonymous version of the protocol separately, the differences with `Protocol II` are highlighted:

- In step 2 of `Protocol II`, the Bobs can retrieve the testing key because they are aware of who $\mathcal{A}$ is. In the fully anonymous setting this is not the case, so another method is needed for $\mathcal{A}$ to communicate the testing key. The `TESTING KEY DISTRIBUTION` subprotocol is used for this.

- In step 7 of `Protocol II`, the partially anonymous version of `ANONYMOUS ERROR CORRECTION` is used, which relies on the Bobs knowing who $\mathcal{A}$ is. This is not the case in the fully anonymous setting, so the fully anonymous version must be used, which has worse performance.

8.3.2 | Analysis

`Protocol II` has similarities with `Protocol I`, but there are important differences. As mentioned in the list of issues (see sec. 8.2.3), the original `ACKA` protocol obtains security by making a statement on ε-closeness of the underlying GHZ state, which is not ideal. `Protocol II` derives the security through a direct statement on the generated key instead.

Indeed, the test in step 6 allows one to obtain ε_s-secrecy of the key without ever having to make a statement on the underlying state. A welcome side-effect of this is that there can be considerably fewer testing rounds, which strongly improves the key rate. This solves issue 2 from the list of issues of `Protocol I` discussed in sec. 8.2.3.

Moreover, it means that the testing rounds can be performed differently than in the `Protocol I`. Instead of separately extracting a $|\mathrm{GHZ}_{m+1}\rangle$ state during `AME` and subsequently verifying the state using `VERIFICATION` afterwards, all measurements are performed at the same time. This is made possible by the fact that the participants know what rounds are testing rounds, so that no public source of randomness has to be used.

Compare this with `Protocol I`: there, the public source of randomness inadvertently instructs the non-participants as well which rounds are the testing

Protocol II - ANONYMOUS CONFERENCE KEY AGREEMENT VERSION 2	
Input:	A pre-shared key between $\mathbf{P}$. Parameters L, p, Q_{tol}, Q_Z.
Goal:	A (larger) pre-shared anonymous secret key between the $\mathbf{P}$.

1: $\mathbf{P}$ run IDENTITY DESIGNATION to establish $\mathcal{A}$ as the sender and the m Bobs $\{\mathcal{B}_i\}$ as the other participants.

2: $\mathcal{A}$ generates the *testing key*: a length-L random bit-string. Each bit equals 1 with probability p, indicating a test round. Using some of the pre-shared key, $\mathcal{A}$ announces the encrypted, compressed testing key. All other nodes announce a random bit-string. The other participants, using some of the pre-shared key, retrieve the testing key.

3: Repeat L times:

   1. The untrusted source distributes a state to the nodes in the network; this state should be $|\text{GHZ}_{\mathcal{N}}\rangle$.
   2. The participants measure their qubit in the Z-basis or X-basis if their testing key is 0 or 1, respectively. The Z-basis measurement outcomes form the raw key, the X-basis measurements are used for verification. All non-participants measure in the X-basis.

4: The testing key is anonymously announced: After the quantum memories of $\mathbf{P}$ and $\bar{\mathbf{P}}$ have decohered, the network runs the Parity subprotocol. $\mathcal{A}$ inputs the testing key, everyone else inputs 0.

5: For every test round, the network uses PARITY to determine if the parity of all their measurement outcomes is 1, indicating a failed test round. From all testing rounds $\mathcal{A}$ computes Q_X, the *observed* X-basis QBER. $\mathcal{A}$ encrypts her input, so only she obtains Q_X.

6: $\mathcal{A}$ determines if $Q_X + \gamma(Q_X) \leqslant Q_{\text{tol}} + \gamma(Q_{\text{tol}})$, where $\gamma(Q_X)$ and $\gamma(Q_{\text{tol}})$ are correctional terms for statistical fluctuations [42, 207]. If this is the case, security is verified; if this is not the case, $\mathcal{A}$ aborts.

7: The participants perform partially ANONYMOUS ERROR CORRECTION, thereby consuming $(1-p)\cdot L\cdot h_2(Q_Z)$ plus n bits of pre-shared key. If any participant finds a discrepancy, the protocol is aborted.

8: The participants perform privacy amplification.

rounds. The non-participants therefore have to announce their measurement results of AME *before* the choice of **Keygen**- or **Verification** round is made, so that they show that they have already performed the measurements and cannot 'cheat' during AME for only the **Keygen** rounds (see the analysis in sec. 8.2.1).

In Protocol I the measurement outcomes of the non-participants are always announced, namely during AME. $\mathcal{A}$ uses these to correct the state, so that the resulting state (before **Keygen** or **Verification**) is the $|\mathrm{GHZ}_{m+1}\rangle$ state. Protocol II takes a different approach, where only the measurement outcomes for the testing rounds are used. Although $\mathcal{A}$ can thus not correct the state during the **Keygen** rounds if it has the incorrect phase (i.e. $|0\ldots0\rangle_{\mathbf{P}} - |1\ldots1\rangle_{\mathbf{P}}$, see the analysis in sec. 8.2.1), the Z-basis correlations are unaffected by these incorrect phases. This means that the raw key will be the same regardless if $\mathcal{A}$ performs the correction or not.

The calculation of the parity for the testing rounds is performed differently as well. In Protocol I, all nodes in the network publicly announce their measurement outcomes so that $\mathcal{A}$ can compute the parity that she subsequently uses to perform the correction. Instead, in Protocol II the network runs PARITY to compute the parity of the measurement outcomes, which is ultimately the only piece of information that $\mathcal{A}$ needs to determine the X-basis QBER Q_X. Furthermore, $\mathcal{A}$ uses a randomized input, so that no one else learns the result of the testing rounds, meaning it cannot be used in any attack by the adversary.

An important side-effect is that there cannot be any leakage of identity during these announcements either. The attack described in the last issue in the list in sec. 8.2.3 is therefore not possible. In this attack a corrupted server distributes $+1$ eigenstates of the X operator, so that all non-participants always measure 0, while the participants announce a bit 1 with probability half: these outcomes are never announced directly, but only the parity of all outcomes is computed and obtained as public knowledge.

It should be noted that this approach to determine the X-basis QBER Q_X is only possible because there are much fewer testing rounds in Protocol II compared to Protocol I. The Parity protocol is somewhat costly to run, but this is remedied by the finite key analysis, which shows that a moderate number of testing rounds suffices to obtain strong security.

Another important detail of this new testing approach is that the participants need to be instructed on which rounds are testing rounds *before* the measurements take place, while the non-participants can only learn the testing key *after* all measurements have taken place. The fact that the non-participants need to learn the testing key, is because they need to know which of their outcomes to use as input for PARITY so that $\mathcal{A}$ can determine Q_X.

Informing the non-participants of the testing key is done using PARITY as well. In Protocol I, the non-participants learn what rounds are the testing rounds only *after* they have already announced their measurement outcomes

(which in that protocol happens during the AME step, before the choice of testing round is even made). In Protocol II such an approach would be impossible: even though there are no 'announcements', the non-participants have to use their measurement outcomes as input for PARITY in the testing rounds. This means that waiting to instruct the non-participants on what are the testing rounds until after they use their outcomes is not possible.

The non-participants can therefore use a targeted approach in which they essentially have the option to act differently during the testing rounds. To prevent the set of colluding parties **C** from performing different measurements during the key generation rounds, the instruction of the testing key is delayed until the qubits of all the non-participants have decohered. This means that, even though they are aware of what rounds are the testing rounds before having to commit to the measurement outcomes, they are forced to have measured their qubits already *before* learning the testing key, so that a different measurement strategy during the key generation rounds is not possible.

In the partially anonymous setting, instructing the participants is done in step 2 by using the pre-shared key. Since testing rounds only occur with probability p, the testing key can be compressed to a length of $L{\cdot}h_2(p)$ [176], so that only a moderate amount of pre-shared key is consumed by encrypting it. In the fully anonymous setting, the TESTING KEY DISTRIBUTION subprotocol is performed instead, which is more costly to run.

Unlike Protocol I, Protocol II includes explicit error correction and privacy amplification steps. These steps have been adapted from normal CKA protocols to not leak anonymity. Interestingly, in the partially anonymous setting this does not reduce the key rate compared to non-anonymous CKA protocols [46, 48]; this will be made more precise for the related methods in chapter 9. Again, the fully anonymous case has a different subprotocol for the error correction, which negatively impacts the key rate.

8.3.3 | Key rates of the protocols

One of the main issues of Protocol I, as listed in sec. 8.2.3, is that it is completely non-robust against noise and other imperfections. Indeed, if any **Verification** round fails, the protocol aborts. On the other hand, Protocol II allows for noise and imperfections by explicitly performing error correction and privacy amplification. The participants characterize the Z-basis QBER Q_Z and the typical X-basis error rate Q_{tol} beforehand. During the protocol the true X-basis error rate Q_X is determined by $\mathcal{A}$, but this is only used to verify that it doesn't exceed the pre-determined X-basis error rate threshold Q_{tol}. Privacy amplification is performed in terms of the pre-determined Q_Z and Q_{tol}, so that the asymptotic key rate of the partially anonymous version of Protocol II results in [48]:

$$r_a = \eta^n \left(1 - h_2(Q_{\text{tol}}) - h_2(Q_Z)\right), \tag{8.3}$$

where η is the transmittance of the link between a single node and the central server. η was taken into account in Pub. **[C]** so that a fair comparison against other protocols is possible; see sec. 8.4 as well.

The privacy amplification could, in principle, be performed in terms of the true X-basis QBER Q_X; if $\mathcal{A}$ has not aborted, it is always lower than Q_{tol}, so that the resulting keyrate would be higher. However, by design Q_X is only available to $\mathcal{A}$; she would have to communicate it to all other participants so that they can perform the adjusted privacy amplification. To ensure anonymity, this has to be performed using e.g. the same steps as distributing the testing key; the keyrate would suffer in both the partially and fully anonymous setting.

In the fully anonymous setting, the asymptotic keyrate is similar but includes an extra penalty [48]:

$$r_a = \kappa\eta^n \left(1 - h_2(Q_{\text{tol}}) - h_2(Q_Z)\right), \tag{8.4}$$

where $0 \leqslant \kappa < 1$ is a factor that represents the extra penalties of the `TESTING KEY DISTRIBUTION` and adapted `ANONYMOUS ERROR CORRECTION` subprotocols:

$$\kappa = \left(1 + \frac{n(n-1)\eta^{n-2} h_2(Q_Z)}{\lfloor \frac{n}{2} \rfloor \left(1 - h_2(Q_{X_B}) - h_2(Q_{Z_B})\right)}\right)^{-1}, \tag{8.5}$$

where X_B and Z_B are the the bi-partite X- and Z-bases QBERs.

Note that in these asymptotic keyrates various terms are unaccounted for, as they vanish with increasing block size L. In the finite setting, a fraction p of the total L rounds is used for testing, so that no key can be generated during those rounds. Moreover, due to this finite sample size the correction for statistical fluctuations ($\gamma(Q_X)$) needs to be included [42, 207]. As will be detailed in chapter 9, ensuring ε_c-correctness and ε_s-secrecy involves a penalty to the total amount of key that can be extracted through privacy amplification. Taking all these into account, for the partially anonymous protocol the maximum length of the secure key that can be extracted from the raw key is:

$$\ell(L) = (1-p)\, L \left[1 - h_2(Q_{\text{tol}} + \gamma(Q_{\text{tol}}))\right] - \log \frac{2(n-1)}{\varepsilon_c} - 2 \log \frac{1}{2\varepsilon_s}, \tag{8.6}$$

where $\ell(L)$ has been written as a function of L to emphasize that the amount of extractable secret key is dependent on the block size.

However, this figure does not account for the pre-shared key that was consumed during the protocol; for a fair comparison this must be taken into account. Revealing the testing key to the participants in step 2 consumes $L \cdot h_2(p)$ bits of pre-shared key. During error correction the error syndrome, of length $(1-p) \cdot L \cdot h_2(Q_Z)$ bits, is encrypted using the pre-shared key. Another n bits of pre-shared key are used during the error correction step to

allow all participants to abort. Subtracting these from $\ell(L)$ gives the *effective* or *net* amount of secret key $\ell_{\text{net}}(L)$; dividing this by L results in the net key rate:

$$r_{\text{net}}(L) = \frac{\ell_{\text{net}}(L)}{L} = (1-p)\left[1 - h_2(Q_{\text{tol}} + \gamma(Q_{\text{tol}})) - h_2(Q_Z)\right] - h_2(p) \\ -\frac{1}{L}\left(\log\frac{2(n-1)}{\varepsilon_c} - 2\log\frac{1}{2\varepsilon_s} - n\right). \tag{8.7}$$

When L increases, the latter terms vanish. Moreover, with a larger L a smaller fraction of testing rounds suffices, so that p can be lower. Note that, however, $r_{\text{net}}(L)$ is not necessarily monotonously decreasing in p; the term $\gamma(Q_{\text{tol}})$ is heavily dependent on the number of testing rounds, and therefore on p. It is hard to optimize for p analytically, and it must be done on a case-by-case basis, taking into consideration the values for L, Q_{tol} and the security parameters. The effect of the value of p is addressed in more detail in sec. 10.2.1 for the protocol introduced in chapter 9, which has a similar parameter.

Since both step 2 and step 7 are different in the fully anonymous version of the protocol, the terms in (8.7) according to these steps are different as well. In fact, `TESTING KEY DISTRIBUTION` does not consume the $L \cdot h_2(p)$ bits of pre-shared key, and `ANONYMOUS ERROR CORRECTION` does not consume the n bits to allow the participants to abort. Therefore, as originally presented in [48], the finite key rate of the fully anonymous version of the protocol is in fact higher than its partial counterpart:

$$r_{\text{net}}^{\text{full}}(L) = \frac{\ell_{\text{net}}^{\text{full}}(L)}{L} = (1-p)\left[1 - h_2(Q_{\text{tol}} + \gamma(Q_{\text{tol}})) - h_2(Q_Z)\right] \\ -\frac{1}{L}\left(\log\frac{2(n-1)}{\varepsilon_c} - 2\log\frac{1}{2\varepsilon_s}\right). \tag{8.8}$$

However, it should be noted that this apparent higher keyrate comes at a penalty that is not reflected in the keyrate: the subprotocols of the fully anonymous version make use of private pairwise channels, which in practice means that all parties in the network need to have pre-shared bi-partite secret keys with all other nodes. Moreover, as noted, the fully anonymous version of the protocol makes use of an adapted form of anonymity.

8.3.4 | Adapted definition of of full anonymity

An intricate detail is that, by running the protocol, the participants learn if the protocol aborts or not. Suppose that they have access to the network parameters, i.e. a characterization of the link-error rates. Since they know the (pre-determined) error rate Q_Z and implicitly have a bound on Q_X^{obs}, they can cross-reference their knowledge of the network with these parameters. In the fully anonymous setting the Bobs are not aware of who else is a

participant, but through the knowledge of the network parameters they can rule out certain nodes to be participants, or learn other information regarding **P**. Indeed, if the protocol does not abort, certain nodes with link-errors that are too high can be ruled out from having participated; at the same time subsets of nodes with low enough link-errors can be ruled out as participants if the protocol *does* abort. Either way, anonymity is not guaranteed from the perspective of honest-but-curious participants. The adapted definition of anonymity Pub. **[C]** solves this by explicitly stating that, from the perspective of the Bobs, the network parameters are symmetrical, or that they do not have access to this information. The presentation there refers to this as *weak* (full) anonymity.

8.4 | Conclusion and discussion on network topology

The protocols introduced in this chapter make it possible to perform conference key agreement in an anonymous fashion. The first protocol, `Protocol I`, was a proof-of-concept that is unsuitable for current or near-future technologies. The second protocol, `Protocol II`, and its fully-anonymous variant improve upon the first protocol by remedying the issues of the original protocol as listed in sec. 8.2.3. There are some important differences between the different protocols, but they are the same in one key point: they all use $|\text{GHZ}_{\mathcal{N}}\rangle$ states distributed over the entire network $\mathcal{N}$.

They are thus protocols that involve *multi-partite entanglement*. Pub. **[C]** ([48]) additionally contains an ACKA protocol that utilizes only bi-partite entanglement, involving many Bell pairs that are shared between all the participants. A partially and fully anonymous bi-partite version is included - essentially the `NOTIFICATION` protocol -, both presented in the supplementary material at Sup. **[sA]**. The efficiencies of `Protocol II` and its fully anonymous version are compared against these bi-partite protocols, to determine if multi-partite entanglement can offer a speed-up compared to Bell pairs. It is harder to distribute an n-partite GHZ state in a network than it is to distribute a Bell pair, because n photons have to be transmitted at the same time instead of a single photon. Hence the transmittance η^n is included in (8.3) and (8.4), to offer a more fair comparison. η is a single-valued representation of the quality of the link between a single node and the central server that represents the likeliness that a photon will successfully be transmitted. It should be noted that η is inverse exponentially dependent on the link length.

It was shown that CKA protocols that utilize multi-partite entanglement can have an operational advantage over bi-partite protocols [40, 48]. Interestingly, Pub. **[C]** showed that this advantage of multi-partite entanglement becomes more pronounced when the anonymity requirement is added, so that multi-partite ACKA protocols can provide key rates that are one or two or-

ders of magnitude higher than the bi-partite counterparts. This advantage occurs for a broad selection of both the total number of nodes n, and link distances (encoded by the transmittance η).

Nevertheless, the distribution of the $|\mathrm{GHZ}_{\mathcal{N}}\rangle$ states by the central server means that all nodes in the network must be connected to it, so that the network is assumed to be a star network. This is a stringent network topology, that makes it harder to implement the protocols. Moreover, and equally important, the protocols dictate some level of *trust* in this central server. Depending on what protocol and what version is implemented, the server is assumed to not share certain network parameters, to not collude with corrupted parties $\mathbf{C}$, or to not distribute different states than the expected $|\mathrm{GHZ}_{\mathcal{N}}\rangle$ states. Especially in these latter two cases, if the adversary has power over the server it can easily break anonymity by effectively stopping the protocol.

Chapter 9 introduces an anonymous conference key agreement scheme that aims to solve this problem. By using a different protocol, the requirement for a central distributing server is dropped: each node in the network has to share a connection with only two other nodes in the network, for a considerably more feasible network topology.

9

Anonymous Conference Key Agreement in Linear Networks

As discussed in sec. 8.4, the protocols presented in that chapter make use of network-wide GHZ states, which implies that all nodes are connected to a central distributing server. This somewhat stringent network topology is both impracticable in real-world networks, and this server has to be provided with some level of trust.

As a different approach, Pub. **[D]** ([46]) introduced another ACKA protocol. This protocol is called `LINEAR ANONYMOUS CONFERENCE KEY AGREEMENT` (`LinACKA`) and makes use of another, less stringent network topology. More specifically, instead of all nodes being connected to a central server, each node is connected to only two other nodes. All nodes are understood to be positioned along a line, so that every node is connected to only its direct neighbours, i.e. to its *left* and its *right* neighbour. Such a linear network topology is therefore also known as a *nearest-neighbour* network, and is less stringent than the star topology.

The protocol allows three special nodes in the network, Alice ($\mathcal{A}$), Bob ($\mathcal{B}$) and Charlie ($\mathcal{C}$), who together form the *participants* $\mathbf{P}$, to create a secret key. They do this in a partial anonymous setting (see sec. 7.5), i.e. they are aware of each others identity, encoded by their position in the line. All other parties in

the network, the *non-participants* $\bar{\mathbf{P}}$, are unaware of the positions of Alice, Bob and Charlie, and remain so during and after the protocol. This holds true both for honest-but-curious non-participants, and for dishonest non-participants that actively deviate from the protocol to try to learn the secure key or the identities of the participants. Still, the adversarial model prohibits the non-participants to collude with each other and perform a combined attack, which is a fair model in the linear network topology.

Because there is no central server, the nodes distribute the necessary entanglement themselves by sharing Bell pairs between every pair of neighbours; these Bell pairs are regarded as a resource for the protocol.

The protocol is divided into three parts, where the first two parts are phrased as subprotocols. In the first part, the nodes in the network perform `STATE PREPARATION`: they use the Bell pairs to create the *network state* $|\mathcal{N}\rangle$, which consists of three separate linear cluster states (see Def. 16); these states arise naturally in the linear network topology. During the second part, the participants anonymously extract a three-body $|\mathrm{GHZ}_{\mathbf{P}}\rangle$ from the network state by performing `GHZ EXTRACTION`. The last part of the protocol consists of measurements that the participants perform to obtain the raw key or assert its security, and the various post-processing steps that result in the secure, anonymous key.

Although the adversarial model prohibits the non-participants from colluding to break anonymity, it should be noted that the security proof of the protocol (i.e. regarding the security of the generated key) does not rest on this assumption. Rather, it provides security under a full, broad adversarial model where any number of non-participants can collude with each other and the adversary.

Pub. **[D]** includes a full finite security analysis, which is presented in this thesis. This results in a rigorous finite key rate that is dependent on the various protocol and security parameters. The performance of the protocol is studied in this chapter by simulating the finite key rate in different scenarios, consisting of more simulations and discussions than originally presented in Pub. **[D]**.

This chapter is structured as follows. In sec. 9.1, the setting of the protocol is made more precise, and the notation that is used in the remainder of the chapter is introduced. Section 9.2 contains the protocol statement, divided into the three different parts. Security and anonymity of the protocol is addressed in sec. 9.3, including a statement of the asymptotic and finite key rate. A discussion regarding the performance of the protocol can be found in sec. 9.4, where it is studied what influence the noise levels and block size have on the finite key rate. Finally, certain other aspects of the protocol are discussed in sec. 9.5, including a potential generalisation to more than three participants. The chapter is concluded in the same section.

Various more technical aspects or details have been deferred to appendices. One step of the protocol involves certain *corrections* that the participants have

to perform on their qubits, which are detailed in chapter F. The technical details of the security proof have been deferred to chapter G. Similarly, the technical details of the anonymity proof have been deferred to chapter H.

9.1 | Protocol and security setting

As explained in the introduction, `LinACKA` assumes a nearest-neighbour topology, i.e. a set of n nodes $\{1, 2, \ldots, n\}$ that are positioned along a line. This allows the nodes, similar to chapter 5, to be indicated relative to each other, so that e.g. the *right* neighbour of node 1 is node 2, and that e.g. node 3 would be the leftmost node of the set $\{3, 4, 8\}$. The participants $\mathbf{P}$, i.e. Alice, Bob and Charlie, are positioned arbitrarily in the line, and their positions are indicated by $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$, respectively. The participants are aware of each other (i.e. they know each other's positions), and w.l.o.g. it is assumed that $\mathcal{A} < \mathcal{B} < \mathcal{C}$, so that Alice is the leftmost participant and Charlie is the rightmost participant. The rest of the nodes in the network are the *non-participants* $\bar{\mathbf{P}}$, and they are not aware what positions the participants have, nor are they even aware if they are e.g. on the right of all participants. The setting is depicted in **Fig.** 9.1, where the three participants have taken arbitrary positions.

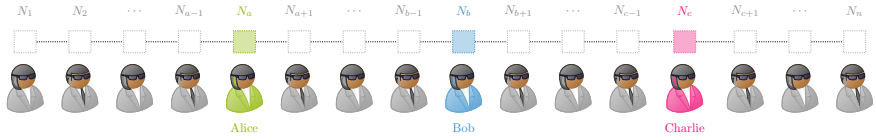


Figure 9.1: Setting of `LinACKA`, where all nodes of the network are positioned along a line. There is no central server, and instead the nodes are connected only to their direct neighbors; therefore the setting is called a *nearest-neighbour* setting. Three special parties, Alice ($\mathcal{A}$), Bob ($\mathcal{B}$) and Charlie ($\mathcal{C}$) aim to establish a shared secret key without the rest of the network learning their identities.

As initial resources, every node is assumed to share an EPR pair with both their left and their right neighbour. Node i thus possess two qubits: one labelled ω_i that is entangled with τ_{i-1}, and one labelled τ_i that is entangled with ω_{i+1}. Because nodes 1 and n both have only one neighbour, they have just one qubit each, τ_i and ω_n, respectively. This initial resource is shown as the top row of **Fig.** 9.2. Like `ACKAv2`, the protocol is a *key-expanding scheme*, so that the participants have access to some pre-shared secret key.

As noted in the introduction to this chapter, it is assumed that the non-participants $\bar{\mathbf{P}}$ are either honest-but-curious or actively deviating from the protocol, but that they do not collude with each other.

9.2 Protocol statement

This section presents `LinACKA`. The three parts of the protocol, where the linear cluster states are created, where the GHZ state is extracted, and where this state is used for key generation or security assertion, are presented separately. The first part is phrased as a subprotocol, `STATE PREPARATION`, and is introduced in sec. 9.2.1. The second part is also phrased as a subprotocol, `GHZ EXTRACTION`, and introduced in sec. 9.2.2. The last part is not stated as a subprotocol but instead explained in sec. 9.2.3. An overview of the first and second step is shown in **Fig.** 9.2. Alternatively, chapter G contains a statement of `LinACKA`, where it is phrased as one complete protocol involving all different parts.

9.2.1 STATE PREPARATION

The first step, the `STATE PREPARATION` subprotocol, aims to create the three linear cluster states from the EPR pairs that are initially distributed. More specifically, by measuring all other qubits, it creates the *network state* $|\mathcal{N}\rangle = |L_{\mathbf{L}}\rangle \otimes |L_{\mathbf{M}}\rangle \otimes |L_{\mathbf{R}}\rangle$, that is composed of the *left*, *middle* and *right* linear cluster states:

$$\begin{aligned} |L_{\mathbf{L}}\rangle &= \left|L_{\tau_1,\tau_2,\ldots,\tau_{\mathcal{A}-1},\omega_{\mathcal{A}}}\right\rangle, \\ |L_{\mathbf{M}}\rangle &= \left|L_{\tau_{\mathcal{A}},\tau_{\mathcal{A}+1},\ldots,\tau_{\mathcal{C}-1},\omega_{\mathcal{C}}}\right\rangle, \\ |L_{\mathbf{R}}\rangle &= \left|L_{\tau_{\mathcal{C}},\tau_{\mathcal{C}+1},\ldots,\tau_{n-1},\omega_n}\right\rangle. \end{aligned} \tag{9.1}$$

Fig. 9.2 shows the network state $|\mathcal{N}\rangle$ in the middle row; the aim of `STATE PREPARATION` is to convert the top row to the middle row.

The protocol consists of three steps, but not all steps are performed by all nodes in the network; **Tab.** 9.1 details what nodes perform what steps. Most notably, a small selection of nodes performs a different second step.

Protocol III - STATE PREPARATION	
Input:	EPR pairs on qubits τ_i and ω_{i+1}.
Goal:	Preparation of the network state $\|\mathcal{N}\rangle$.

All nodes i perform the following steps consecutively:

1: Receive o_{i-1}.
If $o_{i-1} = 1$, apply Z on ω_i.

2a: Perform $C_Z^{(\tau_i,\omega_i)}$ between τ_i and ω_i.
Measure τ_i in X-basis and record measurement outcome bit as o_i.

2b: Draw uniformly random bit o_i.
If $o_i = 1$ apply Z on τ_i.
Apply H on τ_i.

3: Send o_i to $i+1$.

Table 9.1: The table indicates what steps are performed by whom in Protocol III. Nodes 1 and n do not perform step 1 and step 3, respectively, so there is neither an outcome o_0 nor a node $n+1$.

Node	N_1	N_a	N_i	N_c	N_n
1.	✗	✓	✓	✓	✓
2a.	✗	✗	✓	✗	✗
2b.	✓	✓	✗	✓	✗
3.	✓	✓	✓	✓	✗

Note that after the protocol has completed, only Alice and Charlie take part in two different cluster states. All qubits not explicitly stated in (9.1) are measured and removed during the protocol, so that all other nodes only have one qubit left. These qubits can be relabelled: $\tau_i \to i$ for all nodes $\mathcal{N} \setminus \{\mathcal{A}, \mathcal{C}, n\}$ and $\omega_n \to n$. Alice's and Charlie's qubits from the middle linear cluster state are relabelled $\tau_\mathcal{A}, \omega_\mathcal{C} \to \mathcal{A}, \mathcal{C}$, and their qubits from the left and right linear cluster state, respectively, are relabelled $\omega_\mathcal{A}, \tau_\mathcal{C} \to \tilde{\mathcal{A}}, \tilde{\mathcal{C}}$.

As presented, the protocol implicitly assumes that neither Alice nor Charlie are at their respective 'ends' of the linear network. If indeed $\mathcal{A} = 1$, Alice performs those steps according to the 1-column in **Tab.** 9.1. Similarly, if $\mathcal{C} = n$, Charlie performs those steps according to the n-column. Note that in such a case there is no $|L_\mathbf{L}\rangle$ or $|L_\mathbf{R}\rangle$.

9.2.2 GHZ EXTRACTION

The second step, the GHZ EXTRACTION subprotocol, aims to anonymously extract a GHZ state on the participants $\mathbf{P}$ from the network state $|\mathcal{N}\rangle$. Again there are three steps, and similarly to STATE PREPARATION not every step is performed by every node in the network; **Tab.** 9.2 details what nodes perform what steps. Generally speaking, the participants and the non-participants perform different steps, but the outermost nodes 1 and n have their own selection of steps.

In step 2b, the participants perform a *configuration correction* C^i, for $i \in \mathbf{P}$: a local Clifford operation that rotates the post-measurement state of the

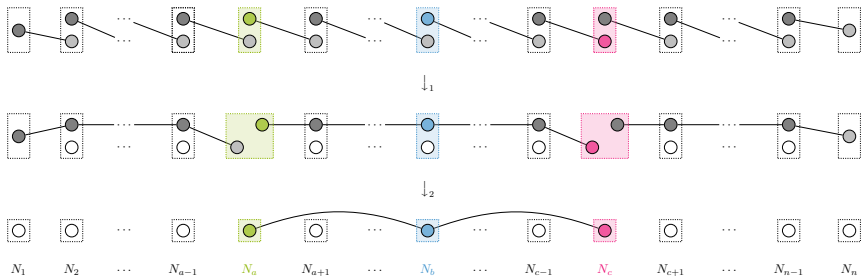


Figure 9.2: (Top): At the start of the protocol, all nodes in the network share a Bell pair with both their left and their right neighbour. **(Middle):** After running `STATE PREPARATION`, the Bell pairs have been consumed to create three linear cluster states, that together form the *network state* $|\mathcal{N}\rangle$. **(Bottom):** Using the network state, the participants extract a $|\mathrm{GHZ}_{\mathbf{P}}\rangle$ state by running `GHZ EXTRACTION`. This state is subsequently used by the participant in either **Keygen** or **Verification** rounds.

participants to the desired GHZ state. These corrections are closely related to those detailed in chapter 5 for the extraction patterns explained there, and are explained in more detail in chapter F; note that they are dependent on the measurement outcomes $\{m_i\}$ of the non-participants.

`Protocol IV - GHZ EXTRACTION`

`Input:` $|\mathcal{N}\rangle$, Corrections $\{C^i\}_{i\in\mathbf{P}}$
`Goal:` Anonymous $|\mathrm{GHZ}_{\mathbf{P}}\rangle$ state.

All nodes i perform the following steps consecutively:

1: Receive bit β_{i-1} and compute $\beta_i = \beta_{i-1} \oplus 1$.

2a: Measure node i in X or Y basis if β_i is 0 or 1, respectively.
Record the measurement outcome bit m_i.

2b: Draw a uniformly random bit m_i.
If $i \in \mathbf{P}$: apply C^i.

3: Communicate β_i to node $i+1$.

Table 9.2:
The table indicates what steps are performed by whom in `Protocol IV`. Node 1 does not perform step 1 but draws a uniformly random bit β_1 and node n does not perform step 3.

Node	N_1	$\mathcal{P}$	$\bar{\mathcal{P}}$	N_n
1.	✗	✓	✓	✓
2a.	✗	✗	✓	✗
2b.	✓	✓	✗	✓
3.	✓	✓	✓	✗

Similarly to `Protocol III`, `Protocol IV` is stated under the implicit assumption that neither Alice nor Charlie are at their respective 'ends' of the

linear network. If indeed $\mathcal{A} = 1$, Alice performs those steps according to the 1-column in **Tab.** 9.2. Similarly, if $\mathcal{C} = n$, Charlie performs those steps according to the n-column.

9.2.3 | Measurements and post-processing

To complete the protocol, the participants use the generated GHZ state either for verification or for key generation, with a probability of p or $1 - p$, respectively, where p is a parameter of the protocol. More specifically, the network runs `STATE PREPARATION` and `GHZ EXTRACTION` a total of L times, referred to as the *block size*. Using $L \cdot h_2(p)$ bits of pre-shared key, the participants divide the L resulting $|\mathrm{GHZ}_{\mathbf{P}}\rangle$ states between $m = \lfloor p \cdot L \rfloor$ randomly chosen **Verification** rounds and $k = L - m$ **Keygen** rounds. For the **Verification** rounds, the participants all measure their qubit in the X basis, recording the measurement outcome, which is used to assert the security of the key.

To allow $\mathcal{A}$ to verify the state during the **Verification** rounds, $\mathcal{B}$ and $\mathcal{C}$ announce their measurement result after each round; every other node in the network announces a random bit to hide their identity. Moreover, so that the non-participants do not have to know what are the **Verification** rounds, all nodes in the network announce a random bit after every **Keygen** round as well. Using the announcements of $\mathcal{B}$ and $\mathcal{C}$, $\mathcal{A}$ computes the fraction of failed **Verification** rounds $Q_X = {}^{(1-\langle X_{\mathcal{A}} X_{\mathcal{B}} X_{\mathcal{C}} \rangle)}/_{2}$. Alice compares this value to a pre-determined tolerance value Q_{tol}, and `REJECTS` when $Q_X \geqslant Q_{\mathrm{tol}}$. In such a case, she sets her *abort bit* to 1, although the actual abort is postponed until a later stage.

The **Keygen** rounds are used by the participants to generate the raw key by measuring their qubits in the Z basis, which results in a raw key of length k in the possession of every participant. However, the participants have to perform *error correction* and *privacy amplification* to ensure that they have an ε_c-correct and ε_s-secret key, respectively.

Although the error correction is comparable to the methods introduced in sec. 7.3.2 and the generalization to more parties in sec. 7.4, there is one main difference. The participants still make use of e.g. an LDPC, so that $\mathcal{A}$ computes the error syndrome e_s of her raw key. However, the error syndrome is not uniformly random, and thus announcing it would give up anonymity, even when this announcement is masked by random announcements from the rest of the network. To effectively hide her identity, $\mathcal{A}$ encrypts it using $|e_s| = k \cdot h_2(Q_Z)$ bits of pre-shared key before announcing it; Q_Z is the maximum (Z-basis) error rate between $\mathcal{A}$ and $\mathcal{B}$, or between $\mathcal{A}$ and $\mathcal{C}$, and is pre-determined. To hide the identity of $\mathcal{A}$, every other node in the network announces the same number of random bits. $\mathcal{B}$ and $\mathcal{C}$, having access to the pre-shared key, are able to decrypt the error syndrome and use it to correct their raw keys $k_{\mathcal{B}}$ and $k_{\mathcal{C}}$.

The verification of the error correction has a similar adaptation. Using

a publicly selected two-universal hashing key, $\mathcal{A}$ calculates the hash $t_{\mathcal{A}}$ of her raw key. To obtain ε_c-correctness, the length of the hash is taken as $|t_{\mathcal{A}}| = \log(1/\varepsilon_c)$, and it is encrypted by Alice using $|t_{\mathcal{A}}|$ bits of pre-shared key to hide her identity; she announces it and every other node in the network announces the same number of random bits. $\mathcal{B}$ and $\mathcal{C}$, after they received and decrypted it using part of the pre-shared key, compare $t_{\mathcal{A}}$ against their own hash and set their respective abort bits to 1 if they do not coincide.

Using three bits of pre-shared key, the participants subsequently announce their encrypted abort bit, while all non-participants announce a random bit. If any of the participant announce the value 1, they abort the protocol. If this is not the case, the participants perform privacy amplification by applying a two-universal hashing function whose output length ℓ is based on the tolerance Q_{tol} and the pre-determined security parameters. The output of this hashing function is the secret key.

9.3 Security and anonymity

The GHZ state is extracted only from the middle linear cluster state $|L_{\mathbf{M}}\rangle$, so effectively only that state is used. The states $|L_{\mathbf{L}}\rangle$ and $|L_{\mathbf{R}}\rangle$ are created as a bi-product during `STATE PREPARATION` - since the non-participants left of $\mathcal{A}$ and right of $\mathcal{C}$ are not aware of their somewhat special position, they take the same steps as those non-participants between $\mathcal{A}$ and $\mathcal{C}$, thereby creating $|L_{\mathbf{L}}\rangle$ and $|L_{\mathbf{R}}\rangle$.

In normal QKD or CKA, the adversary learns the error syndrome and error hash because it is announced through a public channel. Therefore, during privacy amplification, the output of the hashing function is reduced by the upper bound of the amount of information that the adversary can learn from these. As noted in sec. 7.3.3, this is upper bounded by the length of the error syndrome e_s and hash $t_{\mathcal{A}}$, so that length is taken as the amount that needs to be subtracted. However, to guarantee anonymity, both e_s and $t_{\mathcal{A}}$ are encrypted in `LinACKA`[1], so that there is no information leakage during this step; this in turn means that it does not need to be accounted for during privacy amplification either. However, the encryption of e_s and $t_{\mathcal{A}}$ is performed using a pre-shared key, so for a fair comparison it needs to be reduced from the extracted secret key. Interestingly, this anonymous version of error correction does not reduce the key length, as the lengths of e_s and $t_{\mathcal{A}}$ (and thus the amount of consumed pre-shared key) is exactly the amount of information leakage in standard error correction, namely $k \cdot h_2(Q_Z) + \log\left(\frac{1}{\varepsilon_c}\right)$.

An arbitrarily large but finite amount of **Verification** rounds suffices to obtain an arbitrarily good estimate of Q_X, so that in the asymptotic limit p

[1] This is done in the error correction step of `ACKAv2`, `Protocol II` as well, which means a similar argument applies there.

approaches 0. Therefore, all terms in the key rate that are dependent on p vanish as well; this means that the asymptotic key rate becomes:

$$r_a = [1 - h_2(Q_{\text{tol}}) - h_2(Q_Z)], \tag{9.2}$$

where the term $-h_2(Q_{\text{tol}})$ is due to privacy amplification, and the term $-h_2(Q_Z)$ is due to error correction.

In the finite regime, the estimate of Q_X is based on a finite number m of **Verification** rounds. To obtain ε_s-secrecy, a statistical correction $\mu\left(\frac{\varepsilon_s-\varepsilon}{2}, L, p\right)$ is added to the X-basis QBER Q_{tol} (similar to $\gamma(Q_X)$ in (8.7)). This correction depends on the block size L and number of **Verification** rounds m, and introduces a free parameter ε; it is detailed in sec. G.2. This results in a *finite* X-basis QBER estimate $Q_{\text{tol}}^{\text{fin}} = Q_{\text{tol}} + \mu\left(\frac{\varepsilon_s-\varepsilon}{2}, L, p\right) \geqslant Q_{\text{tol}}$. Ultimately, privacy amplification can then output an ε_s-secret key of length $\ell(L) = k\left[1 - h_2(Q_{\text{tol}}^{\text{fin}})\right] - 2\log\left(\frac{1}{\varepsilon}\right)$ (see sec. G.2), which is dependent on the block size L through the statistical correction.

However, a more fair comparison is obtained by reducing the amount of consumed pre-shared key from the output length ℓ. This takes into account the pre-shared key to determine the **Verification** rounds ($L \cdot h_2(p)$ bits), to perform the error correction step ($k \cdot h_2(Q_Z) + \log\left(\frac{1}{\varepsilon_c}\right)$ bits) and to communicate the abort bit (3 bits). Writing $k = L \cdot (1-p)$ and subtracting this pre-shared key results in a *net* secret key rate length:

$$\ell_{\text{net}}(L) = L \cdot (1-p)\left[1 - h_2(Q_{\text{tol}} + \mu\left(\frac{\varepsilon_s - \varepsilon}{2}, L, p\right)) - h_2(Q_Z)\right] - L \cdot h_2(p) \\ -2\log\left(\frac{1}{\varepsilon}\right) + \log\left(\frac{1}{\varepsilon_c}\right) - 3. \tag{9.3}$$

$\varepsilon > 0$ is a free parameter, and p can be freely chosen as well. For given parameters $\varepsilon_s > 0$, $\varepsilon_c > 0$, Q_{tol}, Q_Z and L, $\ell_{\text{net}}(L)$ can thus be optimized over these two parameters. The net finite key rate then is $r_{\text{net}}(L) = \frac{\ell_{\text{net}}(L)}{L}$. The technical details of the proof can be found in chapter G.

Anonymity

There are various steps and details of the protocol that are in place only to guarantee anonymity of the participants. Any non-participant that is immediately to the right of a participant is not aware of their special position, so in step 1 of `Protocol III` they apply a Z correction to their qubit (e.g. $\omega_{\mathcal{A}+1}$) even though there was no measurement outcome (e.g. $o_{\mathcal{A}}$). This is why in step 2b the participants perform the Z operation on their qubit (e.g. $\tau_{\mathcal{A}}$): by randomly applying this operation, they effectively perform an 'anti-correction' which will be corrected by their right neighbour.

Moreover, the alternating (X-Y)-basis measurements during `Protocol IV` are agnostic of the positions of the participants, so that the extraction of the GHZ state can be performed without the positions of the participants leaking. This pattern only works because $\mathcal{A}$ and $\mathcal{C}$ are at their respective ends of the linear cluster state; it is for this reason that the network state $|\mathcal{N}\rangle$ consists of three different cluster states, as prepared during `STATE PREPARATION`.

Similarly to `ACKA` and `ACKAv2`, various announcements of measurement results are announced during `Protocols III` and `IV` (by the selection of nodes as detailed in **Tabs.** 9.1 and 9.2) and the **Verification** and **Keygen** rounds (by $\mathcal{B}$ and $\mathcal{C}$). The nodes that do not have to perform these announcements, announce random bits instead to mask the identity of $\mathcal{B}$ and $\mathcal{C}$. Chapter H gives a detailed proof that the measurement outcomes are indeed uniformly random and uncorrelated, and therefore indistinguishable from these random bits.

However, there is a subtle point to be made: imperfect measurement apparatuses or other noise might alter the probability distribution of the measurement outcomes. In the simple case of measurement-basis agnostic noise (e.g. depolarizing noise), the outcomes are still truly random. However, other types of noise might add a bias to the measurement results that are being announced. This would render the true measurement outcomes distinguishable from the random bits, because the latter do not have such a bias. The nodes in the network can circumvent this by adding such a bias to their announced random bits, but special care needs to be taken to properly mimic the bias that would arise from true measurement outcomes. They can learn this by pre-characterisation of their measurement devices, so that they can simulate the bias accurately. Note that such an adaptation of the measurement outcomes does not affect the key rates or performance of the protocol whatsoever.

9.4 Performance

As with any QKD or CKA protocol, it is essentially assumed that any imperfections in the testing rounds are caused by interference from Eve. These are accounted for during privacy amplification by reducing the secret key length (see the $-h_2(Q_{tol})$ term in (9.2)), but this means that any actual noise will reduce the amount of secret key as well. Moreover, the noise will additionally affect Q_Z, further reducing the total length ℓ. This means that, even in the absence of an adversary, there is a threshold for the QBER, above which no key can be generated. For QKD systems the X-basis and Z-basis QBERs are usually assumed to be equal in this scenario, so that this threshold Q_{thr} becomes the smallest root of $1 - 2 \cdot h_2(Q_{\text{thr}})$, which is $Q_{\text{thr}} \approx 0.11$. Interestingly, the results for `LinACKA` are somewhat different. Because Q_{tol} is a three-party correlation, while Q_Z is a two-party correlation, the latter

can be taken to be $\frac{2}{3}$rd of the former (assuming i.i.d. white noise, to first order)[2]. This means that the threshold Q_{thr} is the first root of the equation $h_2(Q_{\text{thr}}) + h_2(\frac{2}{3}Q_{\text{thr}}) - 1$, which is $Q_{\text{thr}} \approx 0.133$.

The value of ε plays a role in the statistical correction and implicitly determines how many **Verification** rounds are necessary; therefore it sometimes is referred to as ε_{pe}, where *pe* stands for *parameter estimation*. Although an optimization of ε is technically possible, in general its value has little effect on the ultimate key rate r_{net}. Therefore it is often taken that $\varepsilon = \frac{\varepsilon_s}{2}$, so that the two terms in (9.3) that are dependent on it are 'equally distributed'. For all calculations and simulations in this thesis this choice is indeed made.

The value of p has a much greater impact on r_{net} however, so that optimization over p is considerably more important. The choice of p is addressed in more detail in sec. 10.2.1, but can typically be taken $p \leqslant 0.05$, so that only a small fraction of the L states are used for **Verification** rounds. In general, p can be chosen smaller for a larger block size L, because for larger L there are more testing rounds, resulting in a smaller statistical uncertainty in the estimate of Q_X. In the remainder of this chapter, all presented results are optimized over p.

To test the performance of the protocol, the key rates are calculated for various network parameters. **Fig. 9.3** shows the key rate $r_{\text{net}}(L)$ as a function of the block size L, for various noise levels. Note that the key rate can be negative; this means that the post-processing steps consume more key than can be generated, or that there is so much error in the raw key that it can't be corrected efficiently enough. Although the secret key length as presented in (9.3) depends on the tolerance Q_{tol} instead of the actual X-basis QBER, the former can be chosen arbitrarily close to the latter. Hence, in the remainder of this thesis every occurrence of Q_{tol} is replaced by the noise rate Q_X.

From **Fig. 9.3** it is evident that the key rate is heavily dependent on both the block size and the X-basis QBER. The key rate is always monotonically increasing as a function of L; the relative increase is considerably stronger for lower L. This means that for modest block sizes, it is often very useful to continue with the protocol even a little while longer. At the same time, obtaining a positive key rate for low block size is only possible for low error rates; the 'break-even' point, where the key rate first becomes positive, ranges from $L < 5 \times 10^4$ for $Q_X = 0.03$, to as high as $L \sim 3 \times 10^7$ for $Q_X = 0.12$. Additionally, **Fig. 9.3** shows the asymptotic key rate for the lowest included error rate. Remarkably, the block size at which the finite key rate approaches the asymptotic key rate, $L \sim 10 \times 10^{10}$, is similar for all error rate levels.

A more detailed representation is given in **Fig. 9.4**, where the finite key rate r_{net} as a function of both L and Q_X is shown as a surface plot. The main figure on the left depicts r_{net}, while the two smaller graphs on the right depict

[2]Note that, although this gives a higher threshold Q_{thr}, a direct comparison with QKD is unfair: Q_{thr} is a three-party correlation instead of the two-party correlation in QKD, so (for white noise) it will be higher.

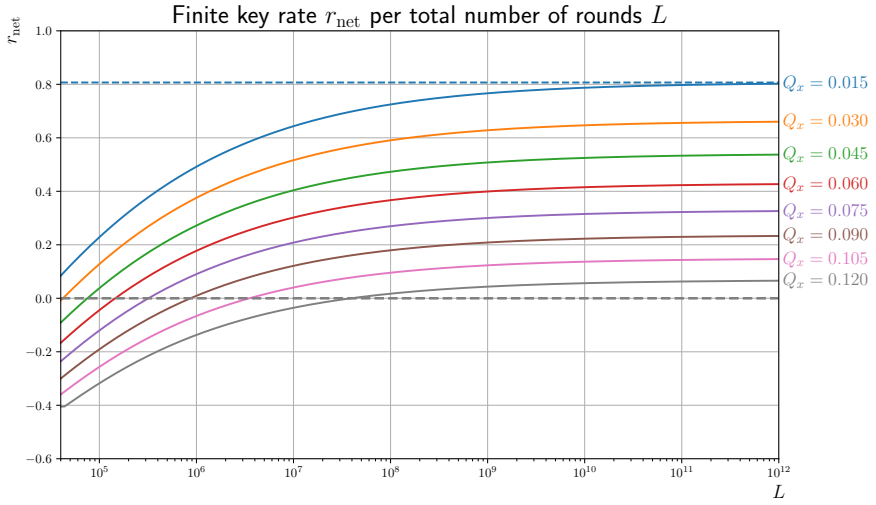


Figure 9.3: Finite key rate r_{net} of `LinACKA` (see (9.3)) as a function of the block size L. The key rates for various X-basis QBERs Q_X are depicted; for every simulation $Q_{\text{tol}} = Q_X$ and the Z-basis QBER is fixed at two-thirds of Q_X to simulate white noise. For the smallest included X-basis QBER ($Q_X = 0.015$) the asymptotic key rate is shown by the dotted line. The security parameters ε_c and ε_s have been fixed at 1×10^{-10}, while $\varepsilon = 5 \times 10^{-11}$, and p is optimized for every block size and noise level individually.

the leading term in (9.3) (the **top right** figure) and the *finite* X-basis QBER $Q_X^{\text{fin}} = Q_X + \mu\left(\frac{\varepsilon_s - \varepsilon}{2}, L, p\right)$ (the **bottom right** figure). The aforementioned threshold Q_{thr} is visible as the blue strip on the right of the main graph, where no positive key rate can be obtained. Similarly, the blue strip on the bottom indicates that, regardless of Q_X, no positive key rate can be obtained for small block size L. This is mostly due to Q_X^{fin} being too large for small L, even if the actual error rate Q_X vanishes.

To offer a separate perspective, suppose that one needs a fixed amount of secret key; for this it useful to know how many network uses (i.e. block size L) are required. **Tab.** 9.3 details this for a selection of different secret key sizes ℓ, for three different X-basis QBER rates. From the table it is evident that for low block sizes L, a small increase in L can have a strong positive effect. Indeed, for e.g. $Q_X = 0.06$ the break-even point is at $L = 2.185 \times 10^5$, but increasing to $L = 2.279 \times 10^5$ already gives 1000 bits of secret key; further increasing to $L = 3.039 \times 10^5$ gives another 9000 additional bits of secret key.

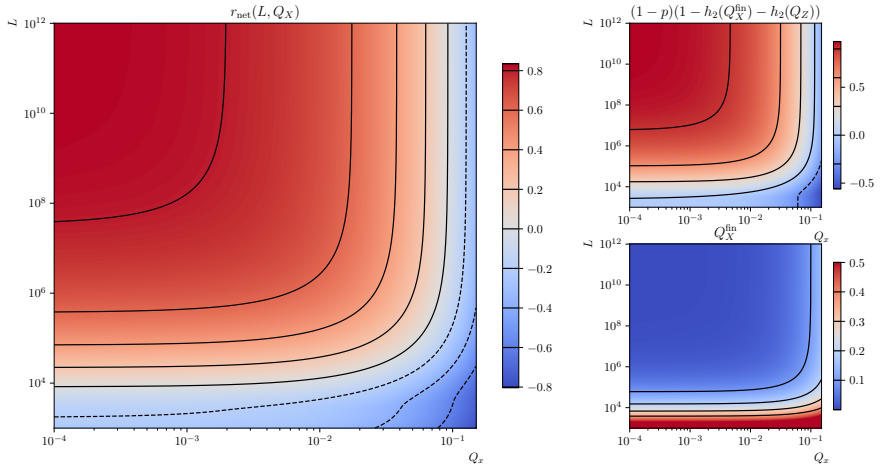


Figure 9.4: The finite key rate r_{net} is shown as a function of the X-basis QBER Q_X and the block size L, and is heavily dependent on both. It is taken that $Q_{\text{tol}} = Q_X$, and the Z-basis QBER has been fixed at $\frac{2}{3}$rds of Q_X to emulate white noise. The key rates have been optimized over p, and the security parameters have all been set to $\varepsilon_c = \varepsilon_s = 1 \times 10^{-10}$, while $\varepsilon = 5 \times 10^{-11}$. **Left:** The net finite key rate r_{net} (see (9.3)) is monotonically increasing with L and monotonically decreasing with Q_X. **Top right:** the leading term in (9.3) closely corresponds to the total amount, but differs for smaller L due to terms in (9.3) that are independent of Q_X. **Bottom right:** the *finite* X-basis QBER $Q_X^{\text{fin}} = Q_X + \mu\left(\frac{\varepsilon_s - \varepsilon}{2}, L, p\right)$ (i.e. including the statistical correction) is strongly increased by a small block size L.

9.5 | Discussion and conclusion

The network state $|\mathcal{N}\rangle$ that is created during `STATE PREPARATION` is technically dependent on the positions of $\mathcal{A}$ and $\mathcal{C}$. However, a straightforward computation reveals that the reduced states of every individual qubit is maximally mixed, so that the quantum state of any node in the network does not contain any information regarding the set of the participants.

As presented, the participants have to apply the correction operators on their qubits before they are able to measure them in either the Z or X basis. This is a highly undesirable property, especially since these corrections are dependent on the measurement outcomes of the non-participants. Waiting until the measurement outcomes are communicated implies that the participants need a quantum memory, which would limit the feasibility of an implementation of the protocol. However, these corrections are of such nature that they can be dealt with differently. All corrections are Clifford operators, so that their ultimate effect is only a potential change of measurement basis

$\log_{10}(\ell)$	$Q_x^m = 0.03$	$Q_x^m = 0.06$	$Q_x^m = 0.10$
0	5.112e+04	2.185e+05	1.358e+07
3	5.696e+04	2.279e+05	1.361e+07
4	9.860e+04	3.039e+05	1.392e+07
5	3.770e+05	8.439e+05	1.679e+07
6	2.411e+06	4.600e+06	3.854e+07

Table 9.3: For fixed secret key lengths ℓ (in base 10 logarithm), the minimum necessary block size L is given. The table details L for three different levels of X-basis QBER Q_X, here denoted Q_X^m. Remember that it is taken $Q_{\text{tol}} = Q_X$, and that Q_Z has been fixed at two-thirds of every Q_X to simulate white noise. The security parameters ε_c and ε_s have been fixed at 1×10^{-8}, $\varepsilon = 5 \times 10^{-9}$, and the simulations are optimized over p from (9.3).

to another Pauli operator. This is also why the corrections, as presented in chapter F, have been explicitly divided between the *configuration correction* and the *measurement outcome dependent corrections*. The configuration corrections involve Clifford operations that might rotate the X- and Z-basis measurements to another Pauli basis, but these correction can be calculated beforehand, as the distance between Alice, Bob and Charlie is known. The measurement outcome dependent corrections are, as detailed in sec. F.1, at most an X and Z operator. The action of these operators on any Pauli-basis measurement is, at most, that the outcome are flipped (e.g. $XZX^\dagger = -Z$, so that the $+1$ and -1 eigenspaces are interchanged, and therefore the measurement outcomes as well). These corrections can be implemented in post-processing, so that the actual measurements can take place before the outcomes of the non-participants are communicated; this removes the need for any quantum memory. This approach is similar to the technique discussed in sec. 5.5, and is exemplary of a broader topic that is discussed in chapter 11.

The `LinACKA` protocol in its complete form starts with Bell pairs as an initial resource that are consumed to create the network state $|\mathcal{N}\rangle$ during `STATE PREPARATION`. However, due to the modular presentation of the complete protocol, this is somewhat separate from the rest of the steps. Indeed, if the network were to obtain or realise the $|\mathcal{N}\rangle$ state in any other way, it could still be used in the subsequent steps of the protocol (provided there is no leakage of identity). Moreover, the left- and right linear cluster states are, as noted before, not strictly necessary for the remainder of the protocol. This means that they could be omitted from any alternative approach to realising the middle linear cluster state. Still, of course, the anonymity of the participants

should be safeguarded, so that any alternative state on the 'outer' nodes does not leak the identity of the participants.

Comparison with ACKAv2

In LinACKA, the non-participants do not have to learn what rounds are the testing rounds, but rather just announce random bits after every round (i.e. both the **Verification** and **Keygen** rounds). Only $\mathcal{B}$ and $\mathcal{C}$ announce an actual measurement result after the **Verification** rounds, so that $\mathcal{A}$ is able to assert the security of the key.

This is in contrast to the approach in ACKAv2 (Protocol II from chapter 8), where the measurement outcomes are not announced publicly. Instead, in that protocol the measurement outcomes are used as input to the PARITY protocol, so that $\mathcal{A}$ can ultimately determine the error rate Q_X without any node having to publicly announce their measurement results. Still, this PARITY subprotocol is inefficient, so that it is only possible to run it for a select number of rounds. In ACKAv2 this is solved by only explicitly performing this for the **Verification** rounds. The drawback to this approach is that the non-participants have to be made aware of the testing rounds, which ultimately allows them to selectively 'play nice' during only the **Verification** rounds, so that they can perform any attack during the **Keygen** rounds without being caught. The approach of ACKAv2 to remedy this, is to wait long enough so that the quantum systems of all nodes have decohered (i.e. the bounded storage model), so that the non-participants are not aware of the type of the round when they perform their measurement.

Nevertheless, the approach from ACKAv2 solves an issue as well. As already noted in Pub. **[A]**, an adversary that has power over the central server can distribute $+1$ eigenstates of the X-basis measurements that the non-participants perform during ACKAv2. Any node that announces the incorrect outcome would then inadvertently give up their identity as a participant. Even though this would result in failed **Verification** rounds, this is effectively 'too late': the identity leakage has already happened. Because in ACKAv2 these outcomes are never announced directly but only used as input to the PARITY protocol, there is no such attack possible.

Essentially, the two approaches can thus be seen as a trade-off between assumptions on the nodes in the network themselves, and assumptions on the power of the adversary over the central server. ACKAv2, defined on a star topology with a strong central server, does not put any limitations on the central server, and therefore uses the bounded storage model to put assumptions on the nodes themselves. LinACKA, on the other hand, does not have a distributing server, and thus puts its assumptions on the power of the adversary: it assumes that the adversary cannot corrupt multiple nodes at once to perform a collective attack involving multiple dishonest nodes.

Collective attacks to break anonymity

Indeed, the restrictions in the adversarial model that is put in place to limit the non-participants to not perform colluding attacks, is put in place only so that anonymity cannot be broken. A straightforward analysis shows that, by actively deviating from the protocol, any pair of nodes $i-1$ and $i+1$ can determine if the node i in between is a participant or not. More specifically, they can perform measurements during `Protocol IV` that are different from the prescribed X- or Y-basis measurements to exploit the stabilizer structure of the linear cluster state. By e.g. both measuring in the Z-basis, they can effectively create a 3-body measurement on $i-1$, i and $i+1$ that is a stabilizer element (i.e. the operator $Z_{(i-1)}X_iZ_{(i+1)}$)[3]. The outcomes of these measurements should be correlated, which can easily be verified by the two colluding nodes because they have access to the outcome m_i. However, in the case that node i is in fact a participant, they would announce a random bit instead of an actual measurement outcome. There is then no correct correlation with 50% probability, from which the colluding nodes $i-1$ and $i+1$ can conclude that node i is a participant. Similarly, if the correlations are always correct, they can conclude that node i is *not* a participant.

It should be noted that this and similar attacks are, in principle, protocol-breaking. By performing these deviations from the protocol, the attacking nodes will affect the state in such a way that ultimately the **Verification** rounds will fail. As such, even in a stronger adversarial model where coherent attacks by colluding parties are allowed, anonymity can not be compromised without such leakage being detected. Furthermore, as noted in the introduction to this chapter, the security proof presented in chapter G (i.e. for the security of the key) does not function under this assumption, but allows for these coherent attacks by colluding parties as well. This means that whenever the protocol does not abort, both security and anonymity are guaranteed.

Generalisation to more than three participants

As presented in Pub. **[D]** ([46]), the protocol specifies that there are exactly three participants. During the protocol, they extract a $|\mathrm{GHZ}_3\rangle$ state from the middle linear cluster state $|L_{\mathbf{M}}\rangle$ as detailed in sec. 9.2.2. However, from the results presented in chapter 5 it follows that a larger GHZ state could be extracted from $|L_{\mathbf{M}}\rangle$. Therefore, in principle a larger set of participants could use this larger GHZ state during **Keygen** and **Verification** rounds to create a shared secret key or assert its security. However, an integral part of the anonymity of `Protocol IV` is the alternating (X-Y)-basis measurement

[3] It is here assumed, for brevity, that node i indeed performs an X-basis measurement instead of a Y-basis measurement. A different attack in the case of a Y-basis measurement is possible as well, but would involve a different selection of colluding nodes.

pattern: using this participant-agnostic pattern, the non-participants cannot determine who are $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$.

The measurement patterns from any larger extraction pattern (i.e. one with more than three participants) would have to be similarly designed so that they are independent of the locations of the participants. Although not trivial to find, such an extraction pattern could be used to increase the number of participants in `LinACKA`. However, note that the corrections that the participants have to perform to obtain the true GHZ state would be less trivial as well, so that a closed form as presented in chapter F might prove impractical.

As pointed out earlier, the measurement pattern from `Protocol IV` with the alternating (X-Y)-basis measurements works because $\mathcal{A}$ and $\mathcal{C}$ are at the ends of a linear cluster state from which the GHZ state is extracted. To realise this, `STATE PREPARATION` creates the network state $|\mathcal{N}\rangle$ where the middle linear cluster state is indeed from $\mathcal{A}$ to $\mathcal{C}$. However, an adapted measurement pattern (for e.g. a larger number of participants) might not necessarily need the participants on such exact positions. Hence, an adapted measurement pattern in `GHZ EXTRACTION` for a larger number of participants might additionally invoke the need of an adaptation in `STATE PREPARATION`, so that e.g. only one, single linear cluster state is created. Still, a measurement pattern for a larger number of participants that is agnostic of the positions has proven difficult to find.

It should be noted that, except for the current presentation of `Protocol IV` and the associated configuration corrections, it is trivial to adapt the protocol to more than three participants. The middle linear cluster state would still be used to extract the (larger) GHZ state, and all measurements, post-processing steps and security- and anonymity proofs are presented in such a way that it is trivial to adapt them to a larger number of participants.

Conclusion

This chapter has introduced `LinACKA`, a protocol to perform anonymous conference key agreement in linear networks. There is no central server that has to distribute the necessary entanglement, which creates a network topology that is less stringent than the star topology of `ACKA` and `ACKAv2` from chapter 8. Both the `ACKA` and the `LinACKA` protocols were implemented in a photonic experimental setup; these implementations are presented and discussed in chapter 10.

10

Experimental Realizations of Anonymous Conference Key Agreement

To complement chapters 8 and 9, this chapter details the collaboration with the experimental group led by Prof. Stephanie Barz at the Universität Stuttgart. They have performed an experimental implementation both of `ACKA` and of `LinACKA`, originally presented in Pubs. **[B]** and **[E]** ([45, 47]), respectively.

The actual experimental implementation of either of the protocols was not performed by me, but by the respective first authors. This chapter focuses on the post-processing of the experimental data, performed by me. It includes both a tomographic analysis of one of the prepared states, and an analysis of the implementation of the protocols themselves, including a calculation of the obtainable key rates for `LinACKA`, specifically.

The first implementation, an experimental realisation of `ACKA` originally presented in Pub. **[B]** ([45]), is presented in sec. 10.1. `LinACKA` is covered by sec. 10.2. More specifically, sec. 10.2.1 contains a detailed discussion on the parameter p, the relative number of **Verification** rounds whose value can greatly influence the finite key rate r_{net}. It was neither included in Pub. **[D]**

nor in Pub. **[E]**, nor presented anywhere else before. Then, sec. 10.2.2 presents the experimental implementation of `LinACKA` from Pub. **[E]**. Some of the post-processing presented in that chapter was not performed by me, but it has been included in the subsection for completeness. Any results that were not obtained or calculated by me are explicitly stated to be so by citing [47]. Most notably, **Fig.** 10.5 is taken (but adapted) from [47]. The chapter is concluded in sec. 10.3.

10.1 | Star network ACKA

In the first experimental realisation, presented in Pub. **[B]** ([45]), a polarisation encoded, four-photon $|\mathrm{GHZ}_4\rangle$ state[1] was prepared in an all-optical setup, with a fidelity of $F = 0.85(\pm 0.02)$ [45]. A tomographic reconstruction of the experimental state can be found in **Fig.** 10.1; for more details on the experimental setup see [45].

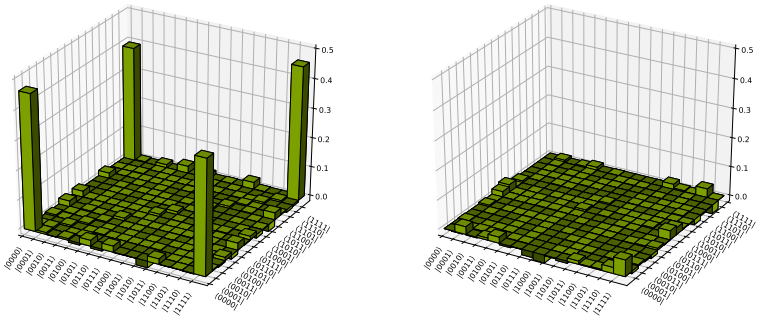


Figure 10.1: State tomographic reconstruction of the experimental state as presented in [45], that is used to implement `ACKA`. The fidelity with the $|\mathrm{GHZ}_4\rangle$ state is $F = 0.85(\pm 0.02)$. A perfect $|\mathrm{GHZ}_4\rangle$ state has four non-zero terms: $|0000\rangle\langle 0000| = |1111\rangle\langle 0000| = |0000\rangle\langle 1111| = |1111\rangle\langle 1111| = \frac{1}{2}$. The experimental state closely resembles this; imperfections and noise are represented by other non-zero entries.

This state acts as the resource for an implementation of `ACKA` in a network of four nodes $\mathcal{N} = \{1, 2, 3, 4\}$. There are six different partitionings $\mathcal{N} = \mathbf{P} \cup \bar{\mathbf{P}}$ chosen for which to implement `ACKA`; four configurations where $|\mathbf{P}| = 3$, labelled $\mathbf{A}-\mathbf{D}$, and two where $|\mathbf{P}| = 2$, labelled $\mathbf{E}-\mathbf{F}$. All these configurations are listed in the first two columns of **Tab.** 10.1.

[1] The prepared experimental state was not exactly the GHZ state, but rather a state LC-equivalent to it. Thus, all of the measurement bases prescribed by the protocol had been rotated as well, but for clarity this is omitted in the presentation in this chapter.

Configuration	P	**Keygen** (success/total)	**Verification** (success/total)	D
A	1, 3, 4	2687/2823	58064/64379	22.81
B	2, 3, 4	1213/1272	23222/26056	20.48
C	1, 2, 3	1210/1265	31111/34924	27.61
D	1, 2, 4	1097/1151	69139/77265	67.13
E	1, 2	2521/2600	67858/75298	28.96
F	3, 4	2647/2749	79047/88202	32.09

Table 10.1: The experimental $|\mathrm{GHZ}_4\rangle$ state is used to implement `ACKA` in 6 different network configurations, i.e. 6 different choices of **P**. They are labelled $\mathbf{A}-\mathbf{F}$, where the second column details the nodes that are in **P**. The successful and total number of **Keygen** and **Verification** rounds are detailed in the third and fourth columns, while the security parameter D, the ratio of number **Verification** to **Keygen** rounds, is given in the last column. Note that the number of participants is 3 for $\mathbf{A}-\mathbf{D}$, while it is 2 for $\mathbf{E}-\mathbf{F}$.

In a complete, networked implementation of `ACKA`, the security parameter D would be specified, and a public source of randomness would be used to determine for every preparation of the $|\mathrm{GHZ}_{\mathcal{N}}\rangle$ state whether it will be used for a **Keygen** or **Verification** round. On the other hand, in the implementation presented here a different approach is taken: all **Keygen** and **Verification** rounds are performed separately in bulk. This results in a total of 12 different measurement settings, two for each network configuration. An artefact of this is that the security parameter D is determined as the ratio of total **Verification** to **Keygen** rounds, instead of vice versa.

Tab. 10.1 details, for every configuration separately, the number of successful and the total number of both **Keygen** and **Verification** rounds. A successful **Verification** round means a round where Alice does not `REJECT`; a successful **Keygen** round means that the outcome of all participants was the same, so that the raw key bits are identical.

In **Fig.** 10.2 the rates for all 12 measurement settings are presented. The results of the **Verification** rounds are an aggregate of all different measurement settings that can arise during the verification rounds. Although the original presentation of `ACKA` in Pub. **[A]** does not include a finite key analysis and as such does not explicitly mention the X- or Z-basis QBER, the rates presented here can be understood as an upper bound on the (inverted) Q_Z (for the **Keygen** rounds) and as the de-facto (inverted) Q_X (for the **Verification** rounds) of the implementation. Note that **Fig.** 10.2 shows the *success rate*, so that a rate of 100% or 0% indicates a QBER of 0 or 1, respectively.

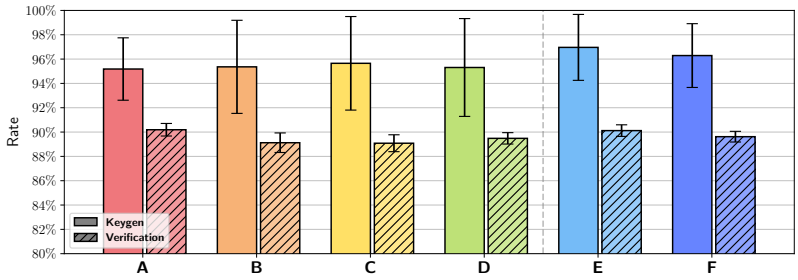


Figure 10.2: Success rates for the experimental implementation of `ACKA`, of both **Keygen** and **Verification** rounds for all 6 different configurations of the network. These rates can be viewed as the de-facto (inverted) Q_Z (for the **Keygen** rounds) and Q_X (for the **Verification** rounds) of the implementation. The results show that all rates lie within the theoretical thresholds, so that an experimental realisation of the protocol with positive keyrates is possible.

The results show that all rates lie within the theoretical thresholds, so that an experimental realisation of the protocol with positive keyrates is possible. The **Keygen** rounds perform considerably better than the **Verification** rounds, which means that the penalties in the key rate due to error correction would be relatively small. This difference in performance is explained by the different types of correlations that are being checked. During the **Verification** rounds, essentially correlations are checked for the entire network (i.e. in this case four-body correlations). The **Keygen** rates on the other hand concern correlations between only the participants. This also explains why the **Keygen** rates are better for configurations **E** and **F** compared to configurations **A** – **D**, because they involve only two-body correlations, instead of three-body correlations.

10.2 | Linear network ACKA

This section discusses implementations of `LinACKA`. Section 10.2.1 discusses the influence of the value of p on the net finite key rate r_{net} (see (9.3)); this was not originally presented in Pub. **[D]** nor anywhere else. Section 10.2.2 presents the experimental implementation of Pub. **[E]**.

10.2.1 | Dependence of finite key rate on p

The parameter p in `LinACKA`, that determines the fraction of rounds that are used for **Verification** rounds, can be chosen freely. However the net secret key rate r_{net} can be strongly dependent on the value of p. There are various terms in the net key rate ((9.3)) that directly depend on it, but some of these terms are additionally dependent on the block size L, the QBER Q_X and the

security parameter ε_s. These parameters are all intricately intertwined, so that the effect of the value of p on the key rate is not always straightforward to understand.

Nevertheless, the value of p can greatly influence the secret key rate: it can make the difference between a negative or positive net key rate r_{net}, or it can change the minimum block size to obtain a fixed r_{net} by orders of magnitude. **Fig.** 10.3 shows a plot of the net key rate r_{net} as a function of p, for various noise levels Q_X.

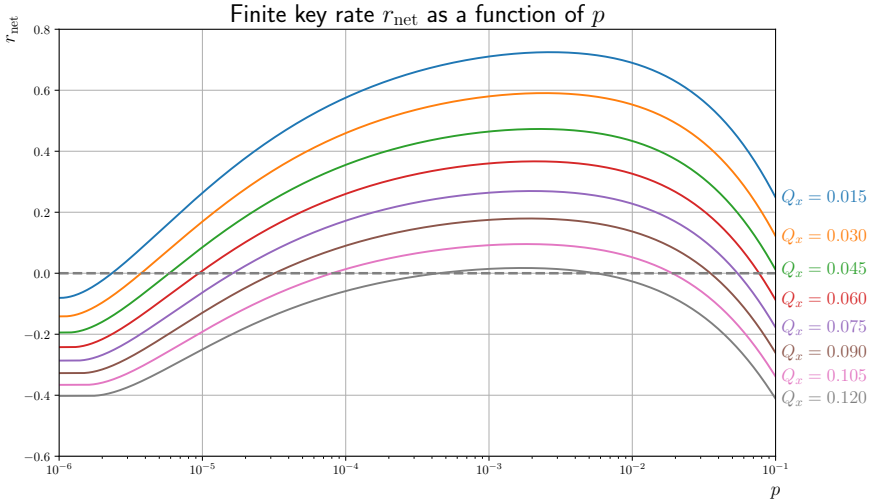


Figure 10.3: The net key rate r_{net} as a function of p, for various Q_X and fixed block size $L = 1 \times 10^8$. It is strongly dependent on p, and for every Q_X there is an optimal choice for p.

The optimal choice of p generally depends on the values of L, Q_X and ε_s, but is not easily found: for fixed values of L, Q_X and the security parameters, (9.3) must be optimized for p. **Fig.** 10.4 shows a 2D plot of the optimal value for p as a function of both L and Q_X.

From **Fig.** 10.4 it is evident that the optimal choice of p is strongly dependent on the block size L. This dependence is clear: p directly determines the (relative) number of **Verification** rounds, but for a moderate block size L this results in a small *absolute* number of **Verification** rounds. Fewer rounds to estimate Q_X results in a larger statistical uncertainty, which has to be accounted for by a larger statistical correction μ, resulting in a shorter secret key.

At the same time, the optimal choice of p is less strongly dependent on the X-basis QBER Q_X. Note that (at a fixed L) the optimal choice of p is lower for a higher Q_X than for a lower Q_X. The reason why is because, in the

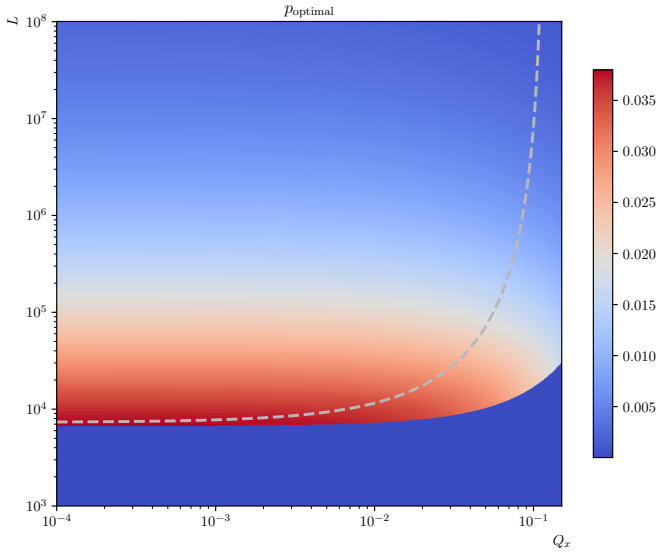


Figure 10.4: Surface plot of p_{optimal}, the choice of p that results in the highest net key rate r_{net}, as a function of both the X-basis QBER Q_X and the block size L. p_{optimal} is heavily dependent on the block size L, especially for smaller sizes. It is less dependent on Q_X, especially in the regime of positive keyrate. The gray dashed line shows the break-even point, under which $r_{\text{net}} < 0$. Note that the L-axis ranges until 1×10^8, unlike 1×10^{12} in e.g. **Fig.** 9.4. The lowest region (where L is smallest) with $p \approx_+ 0$ is explained in the main text.

term $-h_2(Q_X + \mu\left(\frac{\varepsilon_s - \varepsilon}{2}, L, p\right))$ from (9.3), the parameter $\mu = \mu\left(\frac{\varepsilon_s - \varepsilon}{2}, L, p\right)$ is comparatively small when Q_X is high. This means that choosing a smaller p is then less detrimental, because the total term over which the binary entropy is calculated (i.e. $Q_X + \mu$) is already large anyway. It should be noted that the only region where this plays an important effect is under the gray dashed line, i.e. the region where no positive key rate can be obtained anyway.

It can be concluded that in settings with positive net key rate r_{net}, p_{optimal} can be taken roughly independent of Q_X, which is reflected by **Fig.** 10.3. Still, **Fig.** 10.4 shows that it is strongly dependent on the block size L.

For the lowest region in **Fig.** 10.4, the term $-h_2(Q_X + \mu\left(\frac{\varepsilon_s - \varepsilon}{2}, L, p\right))$ in (9.3) becomes so large (i.e. $> \frac{1}{2}$) that it gets cut off regardless of the value of p . Therefore, the lowest p possible optimizes the key rate, because the term $h_2(p)$ in (9.3) is then smallest. However, for this region it holds that $r_{\text{net}} < 0$ for all p anyway, so that choosing the optimal p is irrelevant.

10.2.2 | Experimental implementation of `LinACKA`

For the experimental realisation of `LinACKA` presented in Pub. **[E]** ([47]), a polarisation encoded, four-photon linear cluster state was prepared by fusing two pairs of entangled photons using a photonic C_Z gate [208]. Specifically, a *rotated* $|L_4^{\text{rot}}\rangle$ is prepared from two Bell states $|B_{11}\rangle$ (see (1.52)):

$$\left|L_4^{\text{rot}}\right\rangle = \frac{1}{2}\left(|0101\rangle + |0110\rangle - |1001\rangle + |1010\rangle\right) = C_Z^{(2,3)}\,|B_{11}\rangle \otimes |B_{11}\rangle\,, \quad (10.1)$$

which is related to the $|L_4\rangle$ state by the local Clifford operation $H_1X_2X_3H_4$. Instead of performing this correction, all the measurement bases dictated by the protocol are rotated under this local Clifford operation.

Every photon has its own output mode which ends in a photon detector that clicks when a photon is detected in the mode. The C_Z gate is effectively realised when a click occurs in all four detectors simultaneously, which happens with probability $1/9$ [47]. This means that the realisation of the state is probabilistic but heralded. Hence, the setup is left 'on' for a prolonged time, the *integration time*, during which multiple measurements are aggregated. Therefore, the $|L_4^{\text{rot}}\rangle$ state effectively 'lives' only when it is properly detected and measured. Different measurement bases are realised by including phase shift wave-plates in the setup. For more details of the experimental setup, see [47].

Because of the heralded nature, the number of correct detections (i.e. realisations and measurements of the $|L_4^{\text{rot}}\rangle$ state) is not pre-determined. The fidelity of the experimental state with the $|L_4^{\text{rot}}\rangle$ state was estimated in [47] to be $F = 79.8 \pm 0.8\%$ using state tomography with maximum-likelihood estimation.

In the implementation of the protocol, the nodes $\mathcal{A}$ and $\mathcal{C}$ are fixed to have the first and last qubit of the $|L_4^{\text{rot}}\rangle$ state, respectively. This means that there is a single non-participant, which is either at the second or third qubit. Moreover, `LinACKA` dictates two different measurement bases for this non-participant, namely the X basis ($\beta_2 = 0$) or Y basis ($\beta_2 = 1$). Therefore, there are four different measurement configurations in total, labelled X_2, Y_2, X_3 and Y_3, which are the four different possible measurement operators of the non-participant. The rate of successful **Keygen** and **Verification** rounds for every four of these configurations is presented in **Fig.** 10.5.

The values for the **Keygen** rounds presented in **Fig.** 10.5 are the relative number of 'successful' rounds. *Successful* here indicates that the generated bit for all three participants was equal. This means that the true value for the (inverted) Q_Z will be better (i.e. higher), because this is the maximum of the *pairwise* error rate between $\mathcal{A}$ and the other two participants. It follows that calculating the asymptotic keyrate is, strictly speaking, not possible from the rates presented in **Fig.** 10.5. When uncorrelated errors during the **Keygen** rounds are assumed, the value for Q_Z would, to first order, be $\frac{2}{3}$ of the **Keygen** error rate (i.e. the inversion around 100% from the values presented in

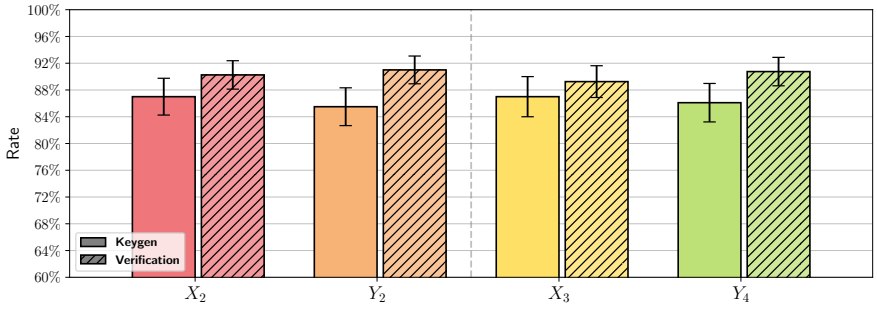


Figure 10.5: This image is taken and adapted from Pub. **[E]** ([47] (Fig. 3)). Success rates for the experimental implementation of `LinACKA`, of both **Keygen** and **Verification** rounds for all 4 different configurations of the network. The rates for the **Keygen** rounds are an upper bound to the (inverted) Q_Z, because they measure the three-party correlations of all participants, instead of the bi-partite correlations that the actual Q_Z reflects. The rates for the **Verification** rounds can be understood as the (inverted) Q_X of the implementation. The results show that all rates lie within the theoretical thresholds, so that an experimental realisation of the protocol with positive keyrates is possible.

Fig. 10.5). Taking this assumption, this would result in a positive asymptotic key rate for all different measurement settings.

For one specific measurement setting, X_2, the experiment was prolonged so that a larger set of rounds was performed. More specifically, there were 10.814 **Keygen** rounds and 294 **Verification** rounds, for a total of 11.108 prepared linear cluster states, resulting in a ratio $p = 0.026$. Out of the 294 **Verification** rounds, there were 33 incorrect measurement outcomes, so that $Q_X = \frac{33}{294} = 0.112$. The bi-partite error rates between $\mathcal{A}$ and $\mathcal{B}$, and $\mathcal{A}$ and $\mathcal{C}$ are $Q_Z^{\mathcal{A},\mathcal{B}} = 0.0959$ and $Q_Z^{\mathcal{A},\mathcal{C}} = 0.0927$, so that the asymptotic key rate of the experimental implementation is:

$$a_r = 1 - h_2(0.112) - h_2(0.0959) = 0.0375. \tag{10.2}$$

The value for Q_Z shows that the above assumption leading to $Q_Z = \frac{2}{3}Q_X$, gives an underestimate for the Z-basis QBER. If the errors were uncorrelated, all bi-partite error rates would be similar, but the bi-partite error rate between $\mathcal{B}$ and $\mathcal{C}$ is $Q_Z^{\mathcal{B},\mathcal{C}} = 0.042$. This is considerably lower than $Q_Z^{\mathcal{A},\mathcal{B}}$ and $Q_Z^{\mathcal{A},\mathcal{C}}$, which is caused by the nature of the preparation of the $|L_4^{\text{rot}}\rangle$ state (see (10.1)). Before the C_Z gate is realised, the state is separable over the bi-partition $\{1,2\} : \{3,4\}$, and only after the entangling gate it becomes a true multi-partite entangled state. This analysis and the relatively low bi-partite error rate $Q_Z^{\mathcal{B},\mathcal{C}}$ shows that the prepared Bell states have a relatively high fidelity, so that the correlations between $\mathcal{B}$ and $\mathcal{C}$ are relatively strong.

It should be noted that the **Verification** rounds are *bunched*, so that multiple **Verification** rounds are performed in succession. More specifically, the total experiment is divided between *runs* of a fixed length of 60 seconds each. A biased random bit generator indicates, at the start of each run, if the rounds in the run will be **Verification** or **Keygen** rounds. The experimental setup is then automatically adapted so that the measurements are performed in the correct basis for that type of round. For each run, the integration time of the setup is thus 60 seconds, during which a random number of correct four-photon clicks occurs. Repetitions of multiple runs then results in an aggregate number of **Verification** rounds and **Keygen** rounds. This practise is common in proof-of-principle experiments, but the choice between **Keygen** and **Verification** is not independent for every round in such an approach, which affects the security of the protocol.

10.3 Conclusion

The experimental implementations that were presented in this chapter have showed that it is possible to utilize multi-partite entanglement in networking protocols. The realisations are proof-of-principle instead of fully-fledged implementations of the complete protocols. Nevertheless, they pave the road forward for quantum networks beyond point-to-point communication.

The presentation of `ACKA` and `LinACKA` in chapters 8 and 9, respectively, is theoretical in nature. `LinACKA` explicitly allows for failed **Verification** and **Keygen** rounds by performing error correction and privacy amplification. Still, the protocol is completely agnostic to the source or type of noise. A better inspection of the typical type of noise that arises in the experimental setup could be beneficial to the performance, and ultimately the keyrates, of the protocols.

Additionally, the theoretical presentation of both protocols assume that the distributed quantum systems are two-level systems, i.e. qubits. The experimental setup from Pubs. **[B]** and **[E]** realises signals that are close to single photons, but as with any experimental implementation there is still a finite probability for other photonic probability distributions and therefore attacks, e.g. the PNS attack. A security proof that takes this into consideration could improve the key rates or increase the security of the protocol implementation. Preferably, such an improved security proof would not have to resort to decoy states (see sec. 7.2.2) or similar techniques, because such measures would reduce the key rates.

Furthermore, the method that is used for cluster state generation is based on a probabilistic gate. This means that a higher number of nodes in the network, which would involve a larger number of C_Z gates, has an exponentially averse effect on the success probability of the preparation of a single network

state. Other methods to realise the entangling gates from the protocol might therefore be necessary.

PART IV

Conclusion

11

Conclusion

The research presented in this thesis has contributed to the advancement of the thriving field of quantum communication and cryptography, by both addressing fundamental questions regarding multi-partite entanglement, and by exploring the utilization of multi-partite entanglement in anonymous quantum networking protocols. Not including the introduction of the relevant basics of quantum information science presented in part I, the thesis was divided into two parts, that covered these two different research areas that I have been active in.

The fundamental questions regarding multi-partite entanglement were the topic of part II, which discussed its distribution, transformation and categorization within quantum networks. Specifically, chapter 5 (Pub. **[F]**) concerned extraction, where we showed that for the exact choice of linear cluster state and GHZ state as resource and target graph states, respectively, the decision of extraction in a network setting is possible. In doing so, we showed an upper bound to the size of any GHZ state that can be extracted from a linear cluster state, and provided a complete characterization of what selections of nodes are possible.

After this, in chapter 6 (Pub. **[G]**), we presented various novel methods to characterize the LU-orbits and entanglement classes of graph states, and novel tools to compare sets of graph states regarding their LU-equivalence. Moreover, we studied the performance of these methods in identifying and distinguishing all LU-orbits and entanglement classes up to nine qubits.

After the part that discussed foundational aspects of multi-partite en-

tanglement, part III dealt with a more operational topic and discussed the utilization of multi-partite entanglement in quantum networking protocols. More specifically, the concept of *anonymity* was introduced in chapter 7, including Defs. 31 and 32, which we originally presented in Pubs. **[A]** and **[C]**, respectively.

In chapter 8 (Pubs. **[A]** and **[C]**), we introduced two protocols to perform anonymous conference key agreement (ACKA), where the first protocol is a proof-of-concept, and the second, more robust, protocol (that has two variants) improves over the first by solving various problems with the first protocol (see sec. 8.2.3). As discussed in sec. 8.4, both these protocols assume a (somewhat restrictive) *star network topology*.

Chapter 9 (Pub. **[D]**) introduced a novel ACKA protocol that assumes a linear network - a less restrictive network topology. In that publication, we showed that ACKA is possible without the presence of a central distributing server, but that the necessary entanglement can be distributed by the nodes of the network themselves. This showed that the utilization of multi-partite entanglement is effective even in network configurations that are more constrained in their resources.

Finally, chapter 10 (Pubs. **[B]** and **[E]**) presented experimental realizations of these ACKA protocols. Although I did not perform these experiments myself, I did perform or aid in the post-processing and the analysis of the experimental data. With these publications, we showed that the utilization of multi-partite entanglement goes beyond theoretical analysis, and that it is possible to utilize it in experimental realisations.

Our research has explored the distribution and categorization of multi-partite entanglement in quantum networks, and has shown that it can be a valuable resource in quantum networking applications. Together with the myriad other recent studies and publications regarding quantum communication and cryptography, our research paves the way forward towards a global quantum internet [32].

Looking ahead, future work could explore many more aspects of both the foundational, theoretical topics of part II, as well as the more operational topics of part III. The next, and final, chapter of this thesis details potential ideas for such future research.

Future research

Some of the conclusions in chapters 5, 6 and 8 to 10, i.e. those presenting my research, include potential ideas for future research that can be conducted. Those ideas are focussed on the topic of each specific chapter, and are largely similar to the ideas presented in the discussions and conclusions of the associated publications. This section takes a broader approach and suggests ideas for future research that do not necessarily fit into the scope of any particular single chapter. Three different topics are addressed, each in their own separate section.

A restriction of the LOCC paradigm

The LOCC paradigm introduced in sec. 4.1 is the de-facto standard for the study of entanglement and the equivalence of states of quantum networks. Most important results in entanglement theory are within the scope of LOCC operations. However, as discussed in sec. 4.1, there are many scenarios where the classical communication that is inherent to the paradigm might not always be practical or possible to perform. As an example, consider a protocol that dictates that a single-qubit correction has to be applied to a node in a network, conditioned on the outcome of a measurement of another node. If the nodes are far enough removed from each other, the time it takes to communicate this outcome might be longer than the decoherence time of the qubit on which the correction operation has to be applied. This would mean that the quantum information is lost before it can be acted upon, and that the protocol would fail.

The LOSR paradigm (see sec. 4.1) is one answer to this problem. Here there is no classical communication possible, but the nodes in the network can rely only on shared randomness. As previously discussed, LOSR is not widely studied, but there are certain no-go results within the paradigm [53, 54].

At the same time, restricting solely to LOSR can be overzealous. Indeed, the techniques and discussions in secs. 5.5 and 9.5 have shown that the conditional gates in the GHZ extraction and in `LinACKA` can be adapted: these corrections are delayed until post-processing, and all necessary quantum operations, including the measurements, can be performed without having to wait for the classical communication to arrive.

Although this approach is not possible for all types of correction operators and measurements of any generic protocol or computation[1], the examples of secs. 5.5 and 9.5 show that there are indeed protocols where it is possible to delay the corrections. It is thus fruitful to consider a 'middle ground' between LOSR and LOCC, where classical communication is not prohibited, but no conditional quantum gates[2] are allowed. In such a paradigm, that I call LODCC (local operations and delayed classical communication), de-cohering quantum systems are much less an issue.

There exists a well-defined mathematical description for LOCC [36], which is easily adapted to describe LOSR. It is unclear if a description of LODCC can be obtained which is equally well-defined.

Towards such a description, note that for the examples from secs. 5.5 and 9.5, the communication can be delayed because the correction operators are single-qubit Pauli operators. These single-qubit Pauli operators at most invert the outcomes of the subsequent Pauli measurements, so their effect can de understood as a re-interpretation of the outcomes. A mathematical description of LODCC would have to reflect this.

This can be extended beyond Pauli measurements and correction operators. Consider the case where the ultimate step of a protocol is described by some POVM $\{E_o\}$. Moreover, suppose that the effect of a correction operator C_m (based on some outcome m of a measurement on another system) at most re-labels the POVM (i.e. the elements of the POVM get 'shuffled around'). Then, the effect of the correction operator can be simulated in post-processing by re-interpreting the measurement outcomes. Beyond mere re-shuffling of the POVM, the correction could be allowed to, for instance, change certain probability distributions associated with the POVM.

Another important open question regarding LODCC, is which known protocols that fall within LOCC do *not* additionally fall under the paradigm of LODCC. I have not been able to find any examples of such protocols.

Multi-partite entanglement in graph states

Part II introduced various aspects of the study of entanglement in networks, with an emphasis on multi-partite entanglement. Many of these results are on the *equivalence* of graph states, where reversible operations are considered that do not involve measurements. As is briefly discussed in sec. 6.7, the methods presented in chapter 6 can potentially be extended to additionally characterize the effect of measurements on graph states.

[1] Indeed, consider measurement-based quantum computation [209]. If any correction operator could be postponed until after the measurements, this would essentially make such quantum computers classically simulable, resulting in the collapse of $\mathbb{BQP}$ to $\mathbb{P}$.

[2] That is, quantum gates that are conditioned on classical data of other systems.

Measurements on graph states and their effects can potentially also be utilized to develop invariants of LU- and LC-orbits or entanglement classes and LC-classes, beyond the marginal dimension (see sec. 6.2). Indeed, consider an n-qubit graph state $|G\rangle$ and its LC-orbit $\mathcal{O}^{\mathrm{LC}}(|G\rangle)$. Additionally, consider the post-measurement states after measuring the first qubit in the X, Y or Z basis, which are $|G_{X_1}\rangle = |\tau_1(\tau_b(G)) \setminus 1\rangle$[3], $|G_{Y_1}\rangle = |\tau_1(G) \setminus 1\rangle$ and $|G_{Z_1}\rangle = |G \setminus 1\rangle$, respectively (see sec. 3.4). Now, because a single-qubit Clifford operation on node 1 can at most permute the three Pauli operators (X_1, Y_1 and Z_1), the set of all three LC-orbits $\{\mathcal{O}^{\mathrm{LC}}(|G_{X_1}\rangle), \mathcal{O}^{\mathrm{LC}}(|G_{Y_1}\rangle), \mathcal{O}^{\mathrm{LC}}(|G_{Z_1}\rangle)\}$ is invariant under local Clifford operations on $|G\rangle$, and can thus act as an identifier of $\mathcal{O}^{\mathrm{LC}}(|G\rangle)$. These three $(n-1)$-qubit LC-orbits can be identified using, for instance, their two-body marginal tensors T_2 (see sec. 6.3), so that different graph states can be compared. Of course, this analysis applies to any node and not just node 1, which leads to n sets of invariants that can all be separately checked for LC-orbits. For LC-classes they can all be combined into a set of n three-tuples.

Preliminary results have shown that this method can distinguish pairs of LC-orbits that cannot be distinguished by the methods of chapter 6. Indeed, the two graphs from **Fig.** 6.7 can be shown to be LC-inequivalent using this method, even though the structure of their marginal dimensions is identical. It should be noted, however, that this method only characterizes equivalence under local Clifford operations, but not local unitary operations.

Note that all results presented in part II consider the specific setting where every node in a network only has access to one qubit of the graph state. Indeed, only local unitary or local Clifford operations are considered, while both Defs. 9 and 10 explicitly allow only single-qubit operations. An interesting extension of the set of allowed operations could include multi-qubit operations like the C_Z gate, but only on specific subsets of nodes[4]. This extension has been investigated in [210], where it is called *party-local Clifford operations*. It is straightforward to show that a C_Z gate between nodes 1 and 2 of a graph state cannot change its marginal dimension $d_{\{1,2\}}$, and it thus seems that the results from chapter 6 can naturally be extended to settings with such party-local Clifford operations. Many questions regarding party-local Clifford operations, or their general unitary counterparts, remain open.

[3] b is some random node in the neighbourhood $\mathcal{N}_1$ of node 1.

[4] Such subsets have to be somewhat restrictive for the problem to be interesting and make sense, and for it to not be too general. Indeed, if a C_Z gate can be applied between any pair of nodes, it is straightforward to see that any graph state can be reproduced from any other graph state, even the empty graph state. Moreover, even when only a 'path' of C_Z gates can be made from any node to any other node, any graph state can be reproduced (this follows from group-theoretic arguments regarding the Clifford group).

Anonymous communication

Part III introduced the topic of anonymity, and gave two definitions, Defs. 31 and 32 (see sec. 7.5). Def. 32 aims to improve over Def. 31 by providing a notion of *approximate* anonymity through the introduction of a security parameter ε_a, but nevertheless problems still persists with this definition. Most notably, open questions remain regarding its composability. First and foremost, it is unclear if a notion of *composable anonymity*, in the spirit of composable security [168], is the correct approach or even an applicable notion.

Indeed, the concept of composable *security* gives guarantees that any *output* of a protocol that is composably secure (i.e. key that was outputted by a QKD protocol) can safely be used in any subsequent application.

The case for anonymity, on the other hand, is different. Def. 32 defines anonymity purely in terms of the protocol, and there is no notion of anonymity associated with the *output* of the protocol, whatever this output may be. As such, the anonymity of the scheme seems to be independent of the output, and therefore independent of the safety of using that output in any subsequent step. In that sense, composability would not be applicable.

Note that, even if a correct notion of composable anonymity can be obtained, Def. 32 would either need to be provide this, or would need to be adapted to do so.

Beyond questions of composability, there exist other issues with the definitions of anonymity. Although Def. 32 does allow for approximate anonymity, the protocols presented in chapters 8 and 9 all obtain 'absolutely anonymity (i.e. $\varepsilon_a = 0$). Although this seems to be a strong feature at first glance, this likely means that small artefacts or discrepancies in an implementation can potentially break anonymity. Such effects can be investigated further. Moreover, the proofs need to potentially be adapted in such a scenario.

Separately, the presented definitions of anonymity were devised in the scope of key generation and broadcasting. However, there are many other applications in quantum communication (see e.g. the introduction of this thesis) for which anonymity in one form or another can be desirable. It remains ambiguous whether anonymity in all these different tasks can be covered by the same definition, or if all these different tasks need their own specific definition of anonymity.

Acknowledgements

Research and science is quintessentially a collaborative effort; none of my scientific publications are by me only. Therefore, I would like to thank all my co-authors, with whom I am grateful to have been able to work.

Specifically, I would like to thank Anna for providing me the opportunity to do my PhD with her at the TUB, for her collaborations and for supervising me throughout these last four years.

Additionally, I would like to thank Frederik, who is my dear friend, made me feel welcome in a covid-struck Berlin, and has been my 'partner in crime' for virtually all my research publications. I would say we make a good team!

My strong gratitude goes out to Ziad, who is not only a dear friend, but also has given me great support, helped me to keep on track with my thesis, has basically proofread it entirely, and has provided me with indispensable feedback.

I would like to thank Nathan for very helpful discussions and answers during the writing process. Furthermore, I would like to thank Fabian, Lina and Marilena for proofreading parts of this thesis, and also giving me invaluable feedback.

Thank you to Anner, Casper, Mats and Stephan for all providing me with an excellent outsiders perspective on my introduction and front matter, thereby giving me the opportunity to try and aim to make this introduction accessible to as broad an audience as possible.

I am indebted to the Python [211] programming language and the NumPy [212] and Matplotlib [213] packages, without which much of my research would not have been possible. Moreover, this thesis has been written using LaTeX, with the (heavily edited) `book` class.

A big thanks to all my colleagues from the QCC group, for all the good times, laughs, dinners, drinks and conferences, and for making my time at the TUB truly memorable.

Last but not least, I would like to thank all my friends and family, who have been my companions throughout the years, and who got me to this point in life.

In het bijzonder gaat dat over mijn ouders, Joselien en Roel. Dankzij hun ben ik wie ik ben, en ik kan nog altijd op hun terugvallen. Dank jullie wel pap & mam.

Het is zo nu en dan ietwat stressvol om een thesis te schrijven, en ondanks dat 'een goede buur beter is dan een verre vriend', is een verre vriend die je

door de mindere dagen heensleept erg waardevol. Daarom: dankjewel lieve Mats.

Dat geldt natuurlijk in het bijzonder ook voor Nina, die sinds twee jaar altijd naast me staat, mij in de gaten houdt, en er voor zorgt dat ik het hoofd koel kan houden. Dankjewel lieve Nina.

PART V

Bibliography

List of publications

Publications and preprints

[A] F. Hahn, J. de Jong, A. Pappa. *Anonymous Conference Key Agreement.* PRX Quantum 1, **020325** (2020), arXiv:2010.04534

[B] C. Thalacker, F. Hahn, J. de Jong, A. Pappa, S. Barz. *Anonymous and secret communications in quantum networks.* New J. Phys. 23 **083026** (2021), arXiv:2103.08722

[C] F. Grasselli[†], G. Murta[†], J. de Jong, F. Hahn, D. Bruß, H. Kampermann, A. Pappa. *Secure Anonymous Conferencing in Quantum Networks.* PRX Quantum 3, **040306** (2022), arXiv:2111.05363

[D] J. de Jong, F. Hahn, J. Eisert, N. Walk, A. Pappa. *Anonymous conference key agreement in linear quantum networks.* Quantum 7, **1117** (2023), arXiv:2205.09169

[E] L. Rückle, J. Budde, J. de Jong, F. Hahn, A. Pappa, S. Barz. *Experimental ACKA using linear cluster states.* Phys. Rev. Research 5, **033222** (2023), arXiv:2207.09487

[F] J. de Jong, F. Hahn, N. Tcholtchev, M. Hauswirth, A. Pappa. *Extracting GHZ states from linear cluster states.* Phys. Rev. Research 6, **013330** (2023), arXiv:2211.16758

[G] L. Vandré[††], J. de Jong[††], F. Hahn, A. Burchardt, O. Gühne, A. Pappa. *Distinguishing graph states by the properties of their marginals.* preprint arXiv:2406.09956

[H] A. Burchardt, J. de Jong, L. Vandré. *Algorithm to Verify Local Equivalence of Stabilizer States.* preprint arXiv:2410.03961[§]

Repositories and supplementary material

[sA] F. Grasselli[†], G. Murta[†], J. de Jong, F. Hahn, D. Bruß, H. Kampermann, A. Pappa. *Supplementary material to "Secure Anonymous Conferencing in Quantum Networks".* PRX Quantum (2022)

[†] These authors contributed equally to this work.
[††] These authors contributed equally to this work.
[§] This publication is not discussed in this thesis.

[sB] J. de Jong, F. Hahn, N. Tcholtchev, M. Hauswirth, A. Pappa. *Supplementary material to "Extracting GHZ states from linear cluster states".* Github (2022)

[sC] J. de Jong. *The* `Graphstabilizer` *python package* Github (2024)

Bibliography

[1] Tomamichel, M. & Leverrier, A. A Largely Self-Contained and Complete Security Proof for Quantum Key Distribution. *Quantum* **1,** 14. doi:10.22331/q-2017-07-14-14 (14th July 2017).

[2] Hahn, F., de Jong, J. & Pappa, A. Anonymous Quantum Conference Key Agreement. *PRX Quantum* **1,** 020325. doi:10.1103/PRXQuantum.1.020325 (22nd Dec. 2020).

[3] Hein, M. *et al.* in *Quantum Computers, Algorithms and Chaos* 115–218 (IOS Press, 2006). doi:10.3254/978-1-61499-018-5-115.

[4] Aaronson, S. *Stephen Wiesner (1942-2021)* Shtetl-Optimized. https://scottaaronson.blog/?p=5730 (2024).

[5] Wiesner, S. Conjugate Coding. *SIGACT News* **15,** 78–88. ISSN: 0163-5700. doi:10.1145/1008908.1008920 (1st Jan. 1983).

[6] Bennett, C. H. & Brassard, G. Quantum Cryptography: Public Key Distribution and Coin Tossing. *Theoretical Computer Science. Theoretical Aspects of Quantum Cryptography – Celebrating 30 Years of BB84* **560,** 7–11. ISSN: 0304-3975. doi:10.1016/j.tcs.2014.05.025 (4th Dec. 2014).

[7] Nielsen, M. A. & Chuang, I. L. *Quantum Computation and Quantum Information: 10th Anniversary Edition* 709 pp. ISBN: 978-1-139-49548-6. doi:10.1017/CBO9780511976667 (Cambridge University Press, 9th Dec. 2010).

[8] Broadbent, A. & Schaffner, C. Quantum Cryptography beyond Quantum Key Distribution. *Des. Codes Cryptogr.* **78,** 351–382. ISSN: 1573-7586. doi:10.1007/s10623-015-0157-4 (1st Jan. 2016).

[9] Bozzio, M., Crépeau, C., Wallden, P. & Walther, P. *Quantum Cryptography beyond Key Distribution: Theory and Experiment* arXiv: 2411.08877. Pre-published.

[10] Lunghi, T. *et al.* Self-Testing Quantum Random Number Generator. *Phys. Rev. Lett.* **114,** 150501. doi:10.1103/PhysRevLett.114.150501 (15th Apr. 2015).

[11] Herrero-Collantes, M. & Garcia-Escartin, J. C. Quantum Random Number Generators. *Rev. Mod. Phys.* **89,** 015004. doi:10.1103/RevModPhys.89.015004 (22nd Feb. 2017).

[12] Mannalatha, V., Mishra, S. & Pathak, A. A Comprehensive Review of Quantum Random Number Generators: Concepts, Classification and the Origin of Randomness. *Quantum Inf Process* **22,** 439. ISSN: 1573-1332. doi:10.1007/s11128-023-04175-y (13th Dec. 2023).

[13] Broadbent, A. & Islam, R. *Quantum Encryption with Certified Deletion* in *Theory of Cryptography* (eds Pass, R. & Pietrzak, K.) (Springer International Publishing, Cham, 2020), 92–122. ISBN: 978-3-030-64381-2. doi:10.1007/978-3-030-64381-2_4.

[14] Hiroka, T., Morimae, T., Nishimaki, R. & Yamakawa, T. *Quantum Encryption with Certified Deletion, Revisited: Public Key, Attribute-Based, and Classical Communication* in *Advances in Cryptology – ASIACRYPT 2021* (eds Tibouchi, M. & Wang, H.) (Springer International Publishing, Cham, 2021), 606–636. ISBN: 978-3-030-92062-3. doi:10.1007/978-3-030-92062-3_21.

[15] Gottesman, D. & Chuang, I. *Quantum Digital Signatures* arXiv: quant-ph/0105032. Pre-published.

[16] Wang, T.-Y., Cai, X.-Q., Ren, Y.-L. & Zhang, R.-L. Security of Quantum Digital Signatures for Classical Messages. *Sci Rep* **5,** 9231. ISSN: 2045-2322. doi:10.1038/srep09231 (18th Mar. 2015).

[17] Pirandola, S. *et al.* Advances in Quantum Cryptography. *Adv. Opt. Photon., AOP* **12,** 1012–1236. ISSN: 1943-8206. doi:10.1364/AOP.361502 (31st Dec. 2020).

[18] Broadbent, A., Fitzsimons, J. & Kashefi, E. *Universal Blind Quantum Computation* in *2009 50th Annual IEEE Symposium on Foundations of Computer Science* 2009 50th Annual IEEE Symposium on Foundations of Computer Science (Oct. 2009), 517–526. doi:10.1109/FOCS.2009.36.

[19] Morimae, T. & Fujii, K. Blind Quantum Computation Protocol in Which Alice Only Makes Measurements. *Phys. Rev. A* **87,** 050301. doi:10.1103/PhysRevA.87.050301 (13th May 2013).

[20] Fitzsimons, J. F. & Kashefi, E. Unconditionally Verifiable Blind Quantum Computation. *Phys. Rev. A* **96,** 012303. doi:10.1103/PhysRevA.96.012303 (5th July 2017).

[21] Fitzsimons, J. F. Private Quantum Computation: An Introduction to Blind Quantum Computing and Related Protocols. *npj Quantum Inf* **3,** 1–11. ISSN: 2056-6387. doi:10.1038/s41534-017-0025-3 (15th June 2017).

[22] Crépeau, C., Gottesman, D. & Smith, A. *Secure Multi-Party Quantum Computation* in *Proceedings of the Thiry-Fourth Annual ACM Symposium on Theory of Computing* (Association for Computing Machinery, New York, NY, USA, 19th May 2002), 643–652. ISBN: 978-1-58113-495-7. doi:10.1145/509907.510000.

[23] Ben-Or, M., Crepeau, C., Gottesman, D., Hassidim, A. & Smith, A. *Secure Multiparty Quantum Computation with (Only) a Strict Honest Majority* in *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)* 2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06) (Oct. 2006), 249–260. doi:10.1109/FOCS.2006.68.

[24] Lipinska, V., Ribeiro, J. & Wehner, S. Secure Multiparty Quantum Computation with Few Qubits. *Phys. Rev. A* **102,** 022405. doi:10.1103/PhysRevA.102.022405 (7th Aug. 2020).

[25] Dulek, Y., Grilo, A. B., Jeffery, S., Majenz, C. & Schaffner, C. *Secure Multi-party Quantum Computation with a Dishonest Majority* in *Advances in Cryptology – EUROCRYPT 2020* (eds Canteaut, A. & Ishai, Y.) (Springer International Publishing, Cham, 2020), 729–758. ISBN: 978-3-030-45727-3. doi:10.1007/978-3-030-45727-3_25.

[26] Christandl, M. & Wehner, S. *Quantum Anonymous Transmissions* in *Advances in Cryptology - ASIACRYPT 2005* (ed Roy, B.) (Springer, Berlin, Heidelberg, 2005), 217–235. ISBN: 978-3-540-32267-2. doi:10.1007/11593447_12.

[27] Degen, C. L., Reinhard, F. & Cappellaro, P. Quantum Sensing. *Rev. Mod. Phys.* **89,** 035002. doi:10.1103/RevModPhys.89.035002 (25th July 2017).

[28] Proctor, T. J., Knott, P. A. & Dunningham, J. A. Multiparameter Estimation in Networked Quantum Sensors. *Phys. Rev. Lett.* **120,** 080501. doi:10.1103/PhysRevLett.120.080501 (21st Feb. 2018).

[29] Azahari, N. S., Harun, N. Z., Chai Wen, C., Ramli, S. N. & Ahmad Zukarnain, Z. *Review of Clock Synchronization in Quantum Communications* in *Proceedings of the 2024 13th International Conference on Software and Computer Applications* (Association for Computing Machinery, New York, NY, USA, 30th May 2024), 350–356. ISBN: 9798400708329. doi:10.1145/3651781.3651834.

[30] Buhrman, H. *et al.* Position-Based Quantum Cryptography: Impossibility and Constructions. *SIAM J. Comput.* **43,** 150–178. ISSN: 0097-5397. doi:10.1137/130913687 (Jan. 2014).

[31] Verduyn Lunel, P. *Quantum Position Verification: Loss-tolerant Protocols and Fundamental Limits* doctoral (Universiteit van Amsterdam, 10th Oct. 2024). ISBN: 9789464735611. https://eprints.illc.uva.nl/id/eprint/2320/ (2024).

[32] Wehner, S., Elkouss, D. & Hanson, R. Quantum Internet: A Vision for the Road Ahead. *Science* **362,** eaam9288. doi:10.1126/science.aam9288 (19th Oct. 2018).

[33] *Quantum Internet Alliance | Building a Global Quantum Internet Made in Europe* Quantum Internet Alliance. https://quantuminternetalliance.org/ (2024).

[34] Bell, J. S. On the Einstein Podolsky Rosen Paradox. *Physics Physique Fizika* **1,** 195–200. doi:10.1103/PhysicsPhysiqueFizika.1.195 (1st Nov. 1964).

[35] Horodecki, R., Horodecki, P., Horodecki, M. & Horodecki, K. Quantum Entanglement. *Rev. Mod. Phys.* **81,** 865–942. ISSN: 0034-6861, 1539-0756. doi:10.1103/RevModPhys.81.865. arXiv: quant-ph/0702225 (17th June 2009).

[36] Watrous, J. *The Theory of Quantum Information* ISBN: 978-1-107-18056-7. doi:10.1017/9781316848142 (Cambridge University Press, Cambridge, 2018).

[37] Bengtsson, I. & Zyczkowski, K. *A Brief Introduction to Multipartite Entanglement* arXiv: 1612.07747. Pre-published.

[38] Horodecki, P., Rudnicki, Ł. & Życzkowski, K. *Multipartite Entanglement* arXiv: 2409.04566. Pre-published.

[39] McCutcheon, W. *et al.* Experimental Verification of Multipartite Entanglement in Quantum Networks. *Nat Commun* **7,** 13251. ISSN: 2041-1723. doi:10.1038/ncomms13251 (9th Nov. 2016).

[40] Epping, M., Kampermann, H., macchiavello, C. & Bruß, D. Multi-Partite Entanglement Can Speed up Quantum Key Distribution in Networks. *New J. Phys.* **19,** 093012. ISSN: 1367-2630. doi:10.1088/1367-2630/aa8487 (Sept. 2017).

[41] Zhang, Z., Shi, R. & Guo, Y. Multipartite Continuous Variable Quantum Conferencing Network with Entanglement in the Middle. *Applied Sciences* **8,** 1312. ISSN: 2076-3417. doi:10.3390/app8081312 (8 Aug. 2018).

[42] Grasselli, F., Kampermann, H. & Bruß, D. Conference Key Agreement with Single-Photon Interference. *New J. Phys.* **21,** 123002. ISSN: 1367-2630. doi:10.1088/1367-2630/ab573e (Dec. 2019).

[43] Murta, G., Grasselli, F., Kampermann, H. & Bruß, D. Quantum Conference Key Agreement: A Review. *Advanced Quantum Technologies* **3,** 2000025. ISSN: 2511-9044. doi:10.1002/qute.202000025 (2020).

[44] Proietti, M., Ho, J., Grasselli, F., Barrow, P., Malik, M. & Fedrizzi, A. Experimental Quantum Conference Key Agreement. *Science Advances* **7,** eabe0395. doi:10.1126/sciadv.abe0395 (4th June 2021).

[45] Thalacker, C., Hahn, F., Jong, J. de, Pappa, A. & Barz, S. Anonymous and Secret Communication in Quantum Networks. *New J. Phys.* **23,** 083026. ISSN: 1367-2630. doi:10.1088/1367-2630/ac1808 (Aug. 2021).

[46] De Jong, J., Hahn, F., Eisert, J., Walk, N. & Pappa, A. Anonymous Conference Key Agreement in Linear Quantum Networks. *Quantum* **7,** 1117. doi:10.22331/q-2023-09-21-1117 (21st Sept. 2023).

[47] Rückle, L., Budde, J., de Jong, J., Hahn, F., Pappa, A. & Barz, S. Experimental Anonymous Conference Key Agreement Using Linear Cluster States. *Phys. Rev. Res.* **5,** 033222. doi:10.1103/PhysRevResearch.5.033222 (2023).

[48] Grasselli, F. *et al.* Secure Anonymous Conferencing in Quantum Networks. *PRX Quantum* **3,** 040306. doi:10.1103/PRXQuantum.3.040306 (2022).

[49] Memmen, J., Eisert, J. & Walk, N. *Advantage of Multi-Partite Entanglement for Quantum Cryptography over Long and Short Ranged Networks* arXiv: 2312.13376. Pre-published.

[50] Webb, J. W. *et al.* Experimental Anonymous Quantum Conferencing. *Optica, OPTICA* **11,** 872–875. ISSN: 2334-2536. doi:10.1364/OPTICA.514362 (20th June 2024).

[51] Tsimakuridze, N. & Gühne, O. Graph States and Local Unitary Transformations beyond Local Clifford Operations. *J. Phys. A: Math. Theor.* **50,** 195302. ISSN: 1751-8113, 1751-8121. doi:10.1088/1751-8121/aa67cd. arXiv: 1611.06938 [quant-ph] (12th May 2017).

[52] Dahlberg, A. & Wehner, S. Transforming Graph States Using Single-Qubit Operations. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **376,** 20170325. doi:10.1098/rsta.2017.0325 (28th May 2018).

[53] Makuta, O., Ligthart, L. T. & Augusiak, R. No Graph State Is Preparable in Quantum Networks with Bipartite Sources and No Classical Communication. *npj Quantum Inf* **9,** 1–5. ISSN: 2056-6387. doi:10.1038/s41534-023-00789-3 (18th Nov. 2023).

[54] Wang, Y.-X., Xu, Z.-P. & Gühne, O. Quantum LOSR Networks Cannot Generate Graph States with High Fidelity. *npj Quantum Inf* **10,** 1–6. ISSN: 2056-6387. doi:10.1038/s41534-024-00806-z (22nd Jan. 2024).

[55] Vandré*, L., de Jong*, J., Hahn, F., Burchardt, A., Gühne, O. & Pappa, A. *Distinguishing Graph States by the Properties of Their Marginals* arXiv: 2406.09956 [quant-ph]. Pre-published.

[56] Shannon, C. E. Communication Theory of Secrecy Systems. *The Bell System Technical Journal* **28,** 656–715. ISSN: 0005-8580. doi:10.1002/j.1538-7305.1949.tb00928.x (Oct. 1949).

[57] Diffie, W. & Hellman, M. New Directions in Cryptography. *IEEE Transactions on Information Theory* **22,** 644–654. ISSN: 1557-9654. doi:10.1109/TIT.1976.1055638 (Nov. 1976).

[58] Rivest, R. L., Shamir, A. & Adleman, L. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. *Commun. ACM* **21,** 120–126. ISSN: 0001-0782. doi:10.1145/359340.359342 (1st Feb. 1978).

[59] Acharya, R. *et al. Quantum Error Correction below the Surface Code Threshold* arXiv: 2408.13687. Pre-published.

[60] Reichardt, B. W. *et al. Demonstration of Quantum Computation and Error Correction with a Tesseract Code* arXiv: 2409.04628. Pre-published.

[61] Shor, P. *Algorithms for Quantum Computation: Discrete Logarithms and Factoring* in *Proceedings 35th Annual Symposium on Foundations of Computer Science* Proceedings 35th Annual Symposium on Foundations of Computer Science (Nov. 1994), 124–134. doi:10.1109/SFCS.1994.365700.

[62] Shor, P. W. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *SIAM J. Comput.* **26,** 1484–1509. ISSN: 0097-5397. doi:10.1137/S0097539795293172 (Oct. 1997).

[63] Renner, R. & Wolf, R. Quantum Advantage in Cryptography. *AIAA Journal* **61,** 1895–1910. ISSN: 0001-1452. doi:10.2514/1.J062267 (May 2023).

[64] Kitagawa, F., Morimae, T., Nishimaki, R. & Yamakawa, T. *Quantum Public-Key Encryption with Tamper-Resilient Public Keys from One-Way Functions* in *Advances in Cryptology – CRYPTO 2024* (eds Reyzin, L. & Stebila, D.) (Springer Nature Switzerland, Cham, 2024), 93–125. ISBN: 978-3-031-68394-7. doi:10.1007/978-3-031-68394-7_4.

[65] Malavolta, G. & Walter, M. *Robust Quantum Public-Key Encryption with Applications to Quantum Key Distribution* arXiv: 2304.02999. Pre-published.

[66] Shettell, N., Hassani, M. & Markham, D. *Private Network Parameter Estimation with Quantum Sensors* arXiv: 2207.14450. Pre-published.

[67] Hassani, M., Scheiner, S., Paris, M. G. A. & Markham, D. *Privacy in Networks of Quantum Sensors* arXiv: 2408.01711. Pre-published.

[68] De Jong, J., Hahn, F., Tcholtchev, N., Hauswirth, M. & Pappa, A. Extracting GHZ States from Linear Cluster States. *Phys. Rev. Res.* **6,** 013330. doi:10.1103/PhysRevResearch.6.013330 (27th Mar. 2024).

[69] Wilde, M. M. *Quantum Information Theory* 1st edition. 672 pp. ISBN: 978-1-107-03425-9. doi:10.1017/CBO9781139525343 (Cambridge University Press, Cambridge, 10th June 2013).

[70] Khatri, S. & Wilde, M. M. *Principles of Quantum Communication Theory: A Modern Approach* arXiv: 2011.04672. Pre-published.

[71] Bengtsson, I. & Zyczkowski, K. *Geometry of Quantum States: An Introduction to Quantum Entanglement* doi:10.1017/CBO9780511535048 (Cambridge University Press, Cambridge, 2006).

[72] Vidick, T. & Wehner, S. *Introduction to Quantum Cryptography* 326 pp. ISBN: 978-1-316-51565-5. doi:10.1017/9781009026208 (Cambridge, United Kingdom ; New York, NY, USA, 14th Sept. 2023).

[73] Gottesman, D. E. *Stabilizer Codes and Quantum Error Correction* PhD thesis (California Institute of Technology, 1997). doi:10.7907/rzr7-dt72.

[74] Terhal, B. M. Quantum Error Correction for Quantum Memories. *Rev. Mod. Phys.* **87,** 307–346. doi:10.1103/RevModPhys.87.307 (7th Apr. 2015).

[75] Gottesman, D. Theory of Fault-Tolerant Quantum Computation. *Phys. Rev. A* **57,** 127–137. doi:10.1103/PhysRevA.57.127 (1st Jan. 1998).

[76] Chamberland, C., Iyer, P. & Poulin, D. Fault-Tolerant Quantum Computing in the Pauli or Clifford Frame with Slow Error Diagnostics. *Quantum* **2,** 43. doi:10.22331/q-2018-01-04-43 (4th Jan. 2018).

[77] Shannon, C. E. A Mathematical Theory of Communication. *The Bell System Technical Journal* **27,** 379–423. ISSN: 0005-8580. doi:10.1002/j.1538-7305.1948.tb01338.x (July 1948).

[78] Renner, R. Security of Quantum Key Distribution. *Int. J. Quantum Inform.* **06,** 1–127. ISSN: 0219-7499. doi:10.1142/S0219749908003256 (Feb. 2008).

[79] Konig, R., Renner, R. & Schaffner, C. The Operational Meaning of Min- and Max-Entropy. *IEEE Transactions on Information Theory* **55,** 4337–4347. ISSN: 1557-9654. doi:10.1109/TIT.2009.2025545 (Sept. 2009).

[80] Tomamichel, M., Colbeck, R. & Renner, R. Duality Between Smooth Min- and Max-Entropies. *IEEE Transactions on Information Theory* **56,** 4674–4681. ISSN: 1557-9654. doi:10.1109/TIT.2010.2054130 (Sept. 2010).

[81] Tomamichel, M. *A Framework for Non-Asymptotic Quantum Information Theory* Doctoral Thesis (ETH Zurich, 2012). doi:10.3929/ethz-a-7356080.

[82] Einstein, A., Podolsky, B. & Rosen, N. Can Quantum-Mechanical Description of Physical Reality Be Considered Complete? *Phys. Rev.* **47,** 777–780. doi:10.1103/PhysRev.47.777 (15th May 1935).

[83] Clauser, J. F., Horne, M. A., Shimony, A. & Holt, R. A. Proposed Experiment to Test Local Hidden-Variable Theories. *Phys. Rev. Lett.* **23,** 880–884. doi:10.1103/PhysRevLett.23.880 (13th Oct. 1969).

[84] Mermin, N. D. Simple Unified Form for the Major No-Hidden-Variables Theorems. *Phys. Rev. Lett.* **65,** 3373–3376. doi:10.1103/PhysRevLett.65.3373 (31st Dec. 1990).

[85] Peres, A. Incompatible Results of Quantum Measurements. *Physics Letters A* **151,** 107–108. ISSN: 0375-9601. doi:10.1016/0375-9601(90)90172-K (1st Dec. 1990).

[86] Palazuelos, C. & Vidick, T. Survey on Nonlocal Games and Operator Space Theory. *Journal of Mathematical Physics* **57,** 015220. ISSN: 0022-2488. doi:10.1063/1.4938052 (12th Jan. 2016).

[87] Goyeneche, D., Alsina, D., Latorre, J. I., Riera, A. & Życzkowski, K. Absolutely Maximally Entangled States, Combinatorial Designs, and Multiunitary Matrices. *Phys. Rev. A* **92,** 032316. doi:10.1103/PhysRevA.92.032316 (15th Sept. 2015).

[88] Heinrich, M. *Answer to "Is the Clifford Group Finite?"* Quantum Computing Stack Exchange. https://quantumcomputing.stackexchange.com/a/13650/8141 (2024).

[89] Perez-Garcia, D., Verstraete, F., Wolf, M. M. & Cirac, J. I. Matrix Product State Representations. *Quantum Info. Comput.* **7,** 401–430. ISSN: 1533-7146. doi:10.26421/QIC7.5-6-1 (1st July 2007).

[90] Fattal, D., Cubitt, T. S., Yamamoto, Y., Bravyi, S. & Chuang, I. L. *Entanglement in the Stabilizer Formalism* arXiv: quant-ph/0406168. Pre-published.

[91] Gottesman, D. *The Heisenberg Representation of Quantum Computers* in *22nd International Colloquium on Group Theoretical Methods in Physics* (International Press Cambridge, MA, July 1998), 32–43.

[92] Aaronson, S. & Gottesman, D. Improved Simulation of Stabilizer Circuits. *Phys. Rev. A* **70,** 052328. doi:10.1103/PhysRevA.70.052328 (30th Nov. 2004).

[93] Calderbank, A. R., Rains, E. M., Shor, P. W. & Sloane, N. J. A. Quantum Error Correction and Orthogonal Geometry. *Phys. Rev. Lett.* **78,** 405–408. doi:10.1103/PhysRevLett.78.405 (20th Jan. 1997).

[94] Kliuchnikov, V., Maslov, D. & Mosca, M. Fast and Efficient Exact Synthesis of Single-Qubit Unitaries Generated by Clifford and T Gates. *QIC* **13,** 607–630. ISSN: 15337146, 15337146. doi:10.26421/QIC13.7-8-4 (7&8 May 2013).

[95] Ni, X., Buerschaper, O. & Van den Nest, M. A Non-Commuting Stabilizer Formalism. *Journal of Mathematical Physics* **56,** 052201. ISSN: 0022-2488. doi:10.1063/1.4920923 (12th May 2015).

[96] Webster, M. A., Brown, B. J. & Bartlett, S. D. The XP Stabiliser Formalism: A Generalisation of the Pauli Stabiliser Formalism with Arbitrary Phases. *Quantum* **6,** 815. doi:10.22331/q-2022-09-22-815 (22nd Sept. 2022).

[97] Fiedler, M. Algebraic Connectivity of Graphs. *Czechoslovak Mathematical Journal* **23,** 298–305. ISSN: 0011-4642 (print). https://dml.cz/handle/10338.dmlcz/101168 (2024) (1973).

[98] Chung, F. *Spectral Graph Theory* American Mathematical Society.

[99] Greenberger, D. M., Horne, M. A. & Zeilinger, A. *Going Beyond Bell's Theorem* arXiv e-prints. Pre-published.

[100] Rossi, M., Huber, M., Bruß, D. & Macchiavello, C. Quantum Hypergraph States. *New J. Phys.* **15,** 113022. ISSN: 1367-2630. doi:10.1088/1367-2630/15/11/113022 (Nov. 2013).

[101] Hein, M., Eisert, J. & Briegel, H. J. Multiparty Entanglement in Graph States. *Phys. Rev. A* **69,** 062311. doi:10.1103/PhysRevA.69.062311 (9th June 2004).

[102] Dür, W., Hartmann, L., Hein, M., Lewenstein, M. & Briegel, H.-J. Entanglement in Spin Chains and Lattices with Long-Range Ising-Type Interactions. *Phys. Rev. Lett.* **94,** 097203. doi:10.1103/PhysRevLett.94.097203 (11th Mar. 2005).

[103] Nielsen, M. A. Conditions for a Class of Entanglement Transformations. *Phys. Rev. Lett.* **83,** 436–439. doi:10.1103/PhysRevLett.83.436 (12th July 1999).

[104] Schmid, D., Rosset, D. & Buscemi, F. The Type-Independent Resource Theory of Local Operations and Shared Randomness. *Quantum* **4,** 262. doi:10.22331/q-2020-04-30-262 (30th Apr. 2020).

[105] Wolfe, E., Pozas-Kerstjens, A., Grinberg, M., Rosset, D., Acín, A. & Navascués, M. Quantum Inflation: A General Approach to Quantum Causal Compatibility. *Phys. Rev. X* **11,** 021043. doi:10.1103/PhysRevX.11.021043 (26th May 2021).

[106] Van den Nest, M., Dehaene, J. & De Moor, B. Invariants of the Local Clifford Group. *Phys. Rev. A* **71,** 022310. doi:10.1103/PhysRevA.71.022310 (16th Feb. 2005).

[107] Verstraete, F., Dehaene, J. & De Moor, B. Normal Forms and Entanglement Measures for Multipartite Quantum States. *Phys. Rev. A* **68,** 012103. doi:10.1103/PhysRevA.68.012103 (15th July 2003).

[108] Van den Nest, M., Dehaene, J. & De Moor, B. Graphical Description of the Action of Local Clifford Transformations on Graph States. *Phys. Rev. A* **69,** 022316. doi:10.1103/PhysRevA.69.022316 (24th Feb. 2004).

[109] Dehaene, J. & Moor, B. D. *The Clifford Group, Stabilizer States, and Linear and Quadratic Operations over GF(2)* arXiv: quant-ph/0304125. Pre-published.

[110] OEIS Foundation Inc. *The On-Line Encyclopedia of Integer Sequences* 2024. https://oeis.org/A123456.

[111] Bouchet, A. An Efficient Algorithm to Recognize Locally Equivalent Graphs. *Combinatorica* **11,** 315–329. ISSN: 1439-6912. doi:10.1007/BF01275668 (1st Dec. 1991).

[112] Bouchet, A. Recognizing Locally Equivalent Graphs. *Discrete Mathematics* **114,** 75–86. ISSN: 0012-365X. doi:10.1016/0012-365X(93)90357-Y (28th Apr. 1993).

[113] Van den Nest, M., Dehaene, J. & De Moor, B. Efficient Algorithm to Recognize the Local Clifford Equivalence of Graph States. *Phys. Rev. A* **70,** 034302. doi:10.1103/PhysRevA.70.034302 (17th Sept. 2004).

[114] Dahlberg, A., Helsen, J. & Wehner, S. Counting Single-Qubit Clifford Equivalent Graph States Is #P-complete. *Journal of Mathematical Physics* **61,** 022202. ISSN: 0022-2488. doi:10.1063/1.5120591 (7th Feb. 2020).

[115] Cabello, A., Danielsen, L. E., López-Tarrida, A. J. & Portillo, J. R. Optimal Preparation of Graph States. *Phys. Rev. A* **83,** 042314. ISSN: 1050-2947, 1094-1622. doi:10.1103/PhysRevA.83.042314 (12th Apr. 2011).

[116] Cabello, A., López-Tarrida, A. J., Moreno, P. & Portillo, J. R. Entanglement in Eight-Qubit Graph States. *Physics Letters A* **373,** 2219–2225. ISSN: 0375-9601. doi:10.1016/j.physleta.2009.04.055 (15th June 2009).

[117] Burchardt, A., Jong, J. de & Vandré, L. *Algorithm to Verify Local Equivalence of Stabilizer States* arXiv: 2410.03961. Pre-published.

[118] Nest, M. V. den, Dehaene, J. & De Moor, B. Local Unitary versus Local Clifford Equivalence of Stabilizer States. *Phys. Rev. A* **71,** 062323. ISSN: 1050-2947, 1094-1622. doi:10.1103/PhysRevA.71.062323. arXiv: quant-ph/0411115 (June 2005).

[119] Zeng, B., Chung, H., Cross, A. W. & Chuang, I. L. Local Unitary versus Local Clifford Equivalence of Stabilizer and Graph States. *Phys. Rev. A* **75,** 032325. doi:10.1103/PhysRevA.75.032325 (19th Mar. 2007).

[120] Ji, Z., Chen, J., Wei, Z. & Ying, M. The LU-LC Conjecture Is False. *Quantum Info. Comput.* **10,** 97–108. ISSN: 1533-7146. doi:10.26421/QIC10.1-2-8 (1st Jan. 2010).

[121] Bouchet, A. Graphic Presentations of Isotropic Systems. *Journal of Combinatorial Theory, Series B* **45,** 58–76. ISSN: 0095-8956. doi:10.1016/0095-8956(88)90055-X (1st Aug. 1988).

[122] Oum, S.-i. Rank-Width and Vertex-Minors. *Journal of Combinatorial Theory, Series B* **95,** 79–100. ISSN: 0095-8956. doi:10.1016/j.jctb.2005.03.003 (1st Sept. 2005).

[123] Dahlberg, A., Helsen, J. & Wehner, S. How to Transform Graph States Using Single-Qubit Operations: Computational Complexity and Algorithms. *Quantum Sci. Technol.* **5,** 045016. ISSN: 2058-9565. doi:10.1088/2058-9565/aba763 (Sept. 2020).

[124] Van den Nest, M., Dür, W., Vidal, G. & Briegel, H. J. Classical Simulation versus Universality in Measurement-Based Quantum Computation. *Phys. Rev. A* **75,** 012337. doi:10.1103/PhysRevA.75.012337 (31st Jan. 2007).

[125] Kraus, B. Local Unitary Equivalence of Multipartite Pure States. *Phys. Rev. Lett.* **104,** 020504. doi:10.1103/PhysRevLett.104.020504 (14th Jan. 2010).

[126] Kraus, B. Local Unitary Equivalence and Entanglement of Multipartite Pure States. *Phys. Rev. A* **82,** 032121. doi:10.1103/PhysRevA.82.032121 (30th Sept. 2010).

[127] De Jong, J. *Jarndejong/ExtractingGHZstatesfromLinearClusterStates* 22nd June 2023. https://github.com/jarndejong/Extracting-GHZ-states-from-Linear-Cluster-States (2024).

[128] Briegel, H. J. & Raussendorf, R. Persistent Entanglement in Arrays of Interacting Particles. *Phys. Rev. Lett.* **86,** 910–913. doi:10.1103/PhysRevLett.86.910 (29th Jan. 2001).

[129] Hahn, F. *Quantum Networks* PhD thesis (Freie Universität Berlin, 2022). http://dx.doi.org/10.17169/refubium-41101.

[130] Burchardt, A. & Hahn, F. *The Foliage Partition: An Easy-to-Compute LC-Invariant for Graph States* arXiv: 2305.07645. Pre-published.

[131] Javadi-Abhari, A. *et al. Quantum Computing with Qiskit* 2024. doi:10.48550/arXiv.2405.08810. arXiv: 2405.08810 [quant-ph].

[132] Quantum, I. B. M. *Ibmq_ cairo v1.0.16* 2021. https://quantum-computing.ibm.com/.

[133] Quantum, I. B. M. *Ibmq_ mumbai v1.5.3* 2021. https://quantum-computing.ibm.com/.

[134] Tiurev, K. & Sørensen, A. S. Fidelity Measurement of a Multiqubit Cluster State with Minimal Effort. *Phys. Rev. Res.* **4,** 033162. doi:10.1103/PhysRevResearch.4.033162 (29th Aug. 2022).

[135] Tóth, G. & Gühne, O. Detecting Genuine Multipartite Entanglement with Two Local Measurements. *Phys. Rev. Lett.* **94,** 060501. doi:10.1103/PhysRevLett.94.060501 (17th Feb. 2005).

[136] Cabello, A., Danielsen, L. E., López-Tarrida, A. J. & Portillo, J. R. *Database of Entanglement in Graph States* (12th Apr. 2011). https://www.ii.uib.no/~larsed/entanglement/ (2024).

[137] Vandré, L. *Distributed Quantum Systems from Graph States to Quantum Networks* (Universität Siegen, 2025). https://uni-siegen.sciebo.de/s/9tKb8oIHueWZwPe.

[138] Adcock, J. C., Morley-Short, S., Dahlberg, A. & Silverstone, J. W. Mapping Graph State Orbits under Local Complementation. *Quantum* **4,** 305. doi:10.22331/q-2020-08-07-305 (7th Aug. 2020).

[139] Merkle, R. C. Secure Communications over Insecure Channels. *Commun. ACM* **21,** 294–299. ISSN: 0001-0782. doi:10.1145/359460.359473 (1st Apr. 1978).

[140] Bruß, D. Optimal Eavesdropping in Quantum Cryptography with Six States. *Phys. Rev. Lett.* **81,** 3018–3021. doi:10.1103/PhysRevLett.81.3018 (5th Oct. 1998).

[141] Bennett, C. H. Quantum Cryptography Using Any Two Nonorthogonal States. *Phys. Rev. Lett.* **68,** 3121–3124. doi:10.1103/PhysRevLett.68.3121 (25th May 1992).

[142] Ekert, A. K. Quantum Cryptography Based on Bell's Theorem. *Phys. Rev. Lett.* **67,** 661–663. doi:10.1103/PhysRevLett.67.661 (5th Aug. 1991).

[143] Bennett, C. H., Brassard, G. & Mermin, N. D. Quantum Cryptography without Bell's Theorem. *Phys. Rev. Lett.* **68,** 557–559. doi:10.1103/PhysRevLett.68.557 (3rd Feb. 1992).

[144] Zapatero, V. *et al.* Advances in Device-Independent Quantum Key Distribution. *npj Quantum Inf* **9,** 1–11. ISSN: 2056-6387. doi:10.1038/s41534-023-00684-x (18th Feb. 2023).

[145] Huttner, B., Imoto, N., Gisin, N. & Mor, T. Quantum Cryptography with Coherent States. *Phys. Rev. A* **51,** 1863–1869. doi:10.1103/PhysRevA.51.1863 (1st Mar. 1995).

[146] Lütkenhaus, N. Security against Individual Attacks for Realistic Quantum Key Distribution. *Phys. Rev. A* **61,** 052304. doi:10.1103/PhysRevA.61.052304 (6th Apr. 2000).

[147] Hwang, W.-Y. Quantum Key Distribution with High Loss: Toward Global Secure Communication. *Phys. Rev. Lett.* **91,** 057901. doi:10.1103/PhysRevLett.91.057901 (1st Aug. 2003).

[148] Wang, X.-B. Beating the Photon-Number-Splitting Attack in Practical Quantum Cryptography. *Phys. Rev. Lett.* **94,** 230503. doi:10.1103/PhysRevLett.94.230503 (16th June 2005).

[149] Lo, H.-K., Ma, X. & Chen, K. Decoy State Quantum Key Distribution. *Phys. Rev. Lett.* **94,** 230504. doi:10.1103/PhysRevLett.94.230504 (16th June 2005).

[150] Scarani, V., Acín, A., Ribordy, G. & Gisin, N. Quantum Cryptography Protocols Robust against Photon Number Splitting Attacks for Weak Laser Pulse Implementations. *Phys. Rev. Lett.* **92,** 057901. doi:10.1103/PhysRevLett.92.057901 (6th Feb. 2004).

[151] Heindel, T. *et al.* Quantum Key Distribution Using Quantum Dot Single-Photon Emitting Diodes in the Red and near Infrared Spectral Range. *New J. Phys.* **14,** 083001. ISSN: 1367-2630. doi:10.1088/1367-2630/14/8/083001 (Aug. 2012).

[152] Cerf, N. J., Lévy, M. & Assche, G. V. Quantum Distribution of Gaussian Keys Using Squeezed States. *Phys. Rev. A* **63,** 052311. doi:10.1103/PhysRevA.63.052311 (18th Apr. 2001).

[153] Gottesman, D. & Preskill, J. Secure Quantum Key Distribution Using Squeezed States. *Phys. Rev. A* **63,** 022309. doi:10.1103/PhysRevA.63.022309 (18th Jan. 2001).

[154] Grosshans, F. & Grangier, P. Continuous Variable Quantum Cryptography Using Coherent States. *Phys. Rev. Lett.* **88,** 057902. doi:10.1103/PhysRevLett.88.057902 (16th Jan. 2002).

[155] Serafini, A. *Quantum Continuous Variables: A Primer of Theoretical Methods* 368 pp. ISBN: 978-1-315-11872-7. doi:10.1201/9781315118727 (CRC Press, Boca Raton, 30th Oct. 2017).

[156] Braunstein, S. L. & van Loock, P. Quantum Information with Continuous Variables. *Rev. Mod. Phys.* **77,** 513–577. doi:10.1103/RevModPhys.77.513 (29th June 2005).

[157] Furrer, F. *et al.* Continuous Variable Quantum Key Distribution: Finite-Key Analysis of Composable Security against Coherent Attacks. *Phys. Rev. Lett.* **109,** 100502. doi:10.1103/PhysRevLett.109.100502 (5th Sept. 2012).

[158] Leverrier, A. Security of Continuous-Variable Quantum Key Distribution via a Gaussian de Finetti Reduction. *Phys. Rev. Lett.* **118,** 200501. doi:10.1103/PhysRevLett.118.200501 (16th May 2017).

[159] Deng, F.-G. & Long, G. L. Bidirectional Quantum Key Distribution Protocol with Practical Faint Laser Pulses. *Phys. Rev. A* **70,** 012311. doi:10.1103/PhysRevA.70.012311 (20th July 2004).

[160] Lucamarini, M. & Mancini, S. Secure Deterministic Communication without Entanglement. *Phys. Rev. Lett.* **94,** 140501. doi:10.1103/PhysRevLett.94.140501 (14th Apr. 2005).

[161] Braunstein, S. L. & Pirandola, S. Side-Channel-Free Quantum Key Distribution. *Phys. Rev. Lett.* **108,** 130502. doi:10.1103/PhysRevLett.108.130502 (30th Mar. 2012).

[162] Lo, H.-K., Curty, M. & Qi, B. Measurement-Device-Independent Quantum Key Distribution. *Phys. Rev. Lett.* **108,** 130503. doi:10.1103/PhysRevLett.108.130503 (30th Mar. 2012).

[163] Pirandola, S., Laurenza, R., Ottaviani, C. & Banchi, L. Fundamental Limits of Repeaterless Quantum Communications. *Nat Commun* **8,** 15043. ISSN: 2041-1723. doi:10.1038/ncomms15043 (26th Apr. 2017).

[164] Azuma, K. *et al.* Quantum Repeaters: From Quantum Networks to the Quantum Internet. *Rev. Mod. Phys.* **95,** 045006. doi:10.1103/RevModPhys.95.045006 (20th Dec. 2023).

[165] Coffman, V., Kundu, J. & Wootters, W. K. Distributed Entanglement. *Phys. Rev. A* **61,** 052306. ISSN: 1050-2947, 1094-1622. doi:10.1103/PhysRevA.61.052306. arXiv: quant-ph/9907047 (10th Apr. 2000).

[166] Mayers, D. Unconditional Security in Quantum Cryptography. *J. ACM* **48,** 351–406. ISSN: 0004-5411. doi:10.1145/382780.382781 (1st May 2001).

[167] Shor, P. W. & Preskill, J. Simple Proof of Security of the BB84 Quantum Key Distribution Protocol. *Phys. Rev. Lett.* **85,** 441–444. doi:10.1103/PhysRevLett.85.441 (10th July 2000).

[168] Ben-Or, M., Horodecki, M., Leung, D. W., Mayers, D. & Oppenheim, J. in *Theory of Cryptography* (ed Kilian, J.) red. by Hutchison, D. *et al.,* 386–406 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2005). ISBN: 978-3-540-24573-5 978-3-540-30576-7. doi:10.1007/978-3-540-30576-7_21.

[169] Gottesman, D. & Lo, H.-K. *Proof of Security of Quantum Key Distribution with Two-Way Classical Communications* arXiv: quant-ph/0105121. Pre-published.

[170] König, R., Renner, R., Bariska, A. & Maurer, U. Small Accessible Quantum Information Does Not Imply Security. *Phys. Rev. Lett.* **98,** 140502. doi:10.1103/PhysRevLett.98.140502 (3rd Apr. 2007).

[171] Canetti, R. *Universally Composable Security: A New Paradigm for Cryptographic Protocols* in *Proceedings 42nd IEEE Symposium on Foundations of Computer Science* Proceedings 42nd IEEE Symposium on Foundations of Computer Science (Oct. 2001), 136–145. doi:10.1109/SFCS.2001.959888.

[172] Maurer, U. *Abstraction in Cryptography* in *Advances in Cryptology - CRYPTO 2009* (ed Halevi, S.) (Springer, Berlin, Heidelberg, 2009), 465–465. ISBN: 978-3-642-03356-8. doi:10.1007/978-3-642-03356-8_27.

[173] Maurer, U. *Constructive Cryptography – A New Paradigm for Security Definitions and Proofs* in *Theory of Security and Applications* (eds Mödersheim, S. & Palamidessi, C.) (Springer, Berlin, Heidelberg, 2012), 33–56. ISBN: 978-3-642-27375-9. doi:10.1007/978-3-642-27375-9_3.

[174] Portmann, C. & Renner, R. *Cryptographic Security of Quantum Key Distribution* arXiv: 1409.3525 [quant-ph]. Pre-published.

[175] Portmann, C. & Renner, R. Security in Quantum Cryptography. *Rev. Mod. Phys.* **94,** 025008. doi:10.1103/RevModPhys.94.025008 (29th June 2022).

[176] Elkouss, D., Martinez-mateo, J. & Martin, V. Information Reconciliation for Quantum Key Distribution. *Quantum Info. Comput.* **11,** 226–238. ISSN: 1533-7146. doi:10.26421/QIC11.3-4-3 (1st Mar. 2011).

[177] Slepian, D. & Wolf, J. Noiseless Coding of Correlated Information Sources. *IEEE Transactions on Information Theory* **19,** 471–480. ISSN: 1557-9654. doi:10.1109/TIT.1973.1055037 (July 1973).

[178] Gallager, R. Low-Density Parity-Check Codes. *IRE Transactions on Information Theory* **8,** 21–28. ISSN: 2168-2712. doi:10.1109/TIT.1962.1057683 (Jan. 1962).

[179] Morello, A. & Mignone, V. DVB-S2: The Second Generation Standard for Satellite Broad-Band Services. *Proceedings of the IEEE* **94,** 210–227. ISSN: 1558-2256. doi:10.1109/JPROC.2005.861013 (Jan. 2006).

[180] Carter, J. L. & Wegman, M. N. Universal Classes of Hash Functions. *Journal of Computer and System Sciences* **18,** 143–154. ISSN: 0022-0000. doi:10.1016/0022-0000(79)90044-8 (1st Apr. 1979).

[181] Wegman, M. N. & Carter, J. L. New Hash Functions and Their Use in Authentication and Set Equality. *Journal of Computer and System Sciences* **22,** 265–279. ISSN: 0022-0000. doi:10.1016/0022-0000(81)90033-7 (1st June 1981).

[182] Vadhan, S. P. Pseudorandomness. *TCS* **7,** 1–336. ISSN: 1551-305X, 1551-3068. doi:10.1561/0400000010 (19th Dec. 2012).

[183] Tomamichel, M., Schaffner, C., Smith, A. & Renner, R. Leftover Hashing Against Quantum Side Information. *IEEE Transactions on Information Theory* **57,** 5524–5535. ISSN: 1557-9654. doi:10.1109/TIT.2011.2158473 (Aug. 2011).

[184] Berta, M., Fawzi, O. & Wehner, S. *Quantum to Classical Randomness Extractors* in *Advances in Cryptology – CRYPTO 2012* (eds Safavi-Naini, R. & Canetti, R.) (Springer, Berlin, Heidelberg, 2012), 776–793. ISBN: 978-3-642-32009-5. doi:10.1007/978-3-642-32009-5_45.

[185] Foreman, C. & Masanes, L. *Seedless Extractors for Device-Independent Quantum Cryptography* arXiv: 2403.04713. Pre-published.

[186] Santha, M. & Vazirani, U. V. Generating Quasi-Random Sequences from Semi-Random Sources. *Journal of Computer and System Sciences* **33,** 75–87. ISSN: 0022-0000. doi:10.1016/0022-0000(86)90044-9 (1st Aug. 1986).

[187] Curty, M., Xu, F., Cui, W., Lim, C. C. W., Tamaki, K. & Lo, H.-K. Finite-Key Analysis for Measurement-Device-Independent Quantum Key Distribution. *Nat Commun* **5,** 3732. ISSN: 2041-1723. doi:10.1038/ncomms4732 (29th Apr. 2014).

[188] Grasselli, F. in *Quantum Cryptography: From Key Distribution to Conference Key Agreement* (ed Grasselli, F.) 55–70 (Springer International Publishing, Cham, 2021). ISBN: 978-3-030-64360-7. doi:10.1007/978-3-030-64360-7_4.

[189] Tomamichel, M., Colbeck, R. & Renner, R. A Fully Quantum Asymptotic Equipartition Property. *IEEE Trans. Inf. Theor.* **55,** 5840–5847. ISSN: 0018-9448. doi:10.1109/TIT.2009.2032797 (1st Dec. 2009).

[190] Christandl, M., König, R. & Renner, R. Postselection Technique for Quantum Channels with Applications to Quantum Cryptography. *Phys. Rev. Lett.* **102,** 020504. doi:10.1103/PhysRevLett.102.020504 (14th Jan. 2009).

[191] Dupuis, F., Fawzi, O. & Renner, R. Entropy Accumulation. *Commun. Math. Phys.* **379,** 867–913. ISSN: 1432-0916. doi:10.1007/s00220-020-03839-5 (1st Nov. 2020).

[192] Metger, T. & Renner, R. Security of Quantum Key Distribution from Generalised Entropy Accumulation. *Nat Commun* **14,** 5272. ISSN: 2041-1723. doi:10.1038/s41467-023-40920-8 (29th Aug. 2023).

[193] Metger, T., Fawzi, O., Sutter, D. & Renner, R. *Generalised Entropy Accumulation* in *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)* 2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS) (Oct. 2022), 844–850. doi:10.1109/FOCS54457.2022.00085.

[194] Tomamichel, M. & Renner, R. Uncertainty Relation for Smooth Entropies. *Phys. Rev. Lett.* **106,** 110506. doi:10.1103/PhysRevLett.106.110506 (16th Mar. 2011).

[195] Coles, P. J., Berta, M., Tomamichel, M. & Wehner, S. Entropic Uncertainty Relations and Their Applications. *Rev. Mod. Phys.* **89,** 015002. doi:10.1103/RevModPhys.89.015002 (6th Feb. 2017).

[196] Heisenberg, W. Über den anschaulichen Inhalt der quantentheoretischen Kinematik und Mechanik. *Z. Physik* **43,** 172–198. ISSN: 0044-3328. doi:10.1007/BF01397280 (1st Mar. 1927).

[197] Cabello, A. *Multiparty Key Distribution and Secret Sharing Based on Entanglement Swapping* arXiv: quant-ph/0009025. Pre-published.

[198] Zhou, J. & Guo, Y. Continuous-Variable Measurement-Device Independent Multipartite Quantum Communication Using Coherent States. *J. Phys. Soc. Jpn.* **86,** 024003. ISSN: 0031-9015. doi:10.7566/JPSJ.86.024003 (15th Feb. 2017).

[199] Ottaviani, C., Lupo, C., Laurenza, R. & Pirandola, S. Modular Network for High-Rate Quantum Conferencing. *Commun Phys* **2,** 1–6. ISSN: 2399-3650. doi:10.1038/s42005-019-0209-6 (30th Sept. 2019).

[200] Pappa, A., Chailloux, A., Wehner, S., Diamanti, E. & Kerenidis, I. Multipartite Entanglement Verification Resistant against Dishonest Parties. *Phys. Rev. Lett.* **108,** 260502. doi:10.1103/PhysRevLett.108.260502 (26th June 2012).

[201] Brassard, G., Broadbent, A., Fitzsimons, J., Gambs, S. & Tapp, A. *Anonymous Quantum Communication* in *Advances in Cryptology – ASIACRYPT 2007* (ed Kurosawa, K.) (Springer, Berlin, Heidelberg, 2007), 460–473. ISBN: 978-3-540-76900-2. doi:10.1007/978-3-540-76900-2_28.

[202] Hillery, M., Bužek, V. & Berthiaume, A. Quantum Secret Sharing. *Phys. Rev. A* **59,** 1829–1834. doi:10.1103/PhysRevA.59.1829 (1st Mar. 1999).

[203] Cleve, R., Gottesman, D. & Lo, H.-K. How to Share a Quantum Secret. *Phys. Rev. Lett.* **83,** 648–651. doi:10.1103/PhysRevLett.83.648 (19th July 1999).

[204] Gottesman, D. Theory of Quantum Secret Sharing. *Phys. Rev. A* **61,** 042311. doi:10.1103/PhysRevA.61.042311 (16th Mar. 2000).

[205] Broadbent, A. & Tapp, A. in *Advances in Cryptology – ASIACRYPT 2007* (ed Kurosawa, K.) 410–426 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2007). ISBN: 978-3-540-76899-9. doi:10.1007/978-3-540-76900-2_25.

[206] Damgard, I. B., Fehr, S., Salvail, L. & Schaffner, C. *Cryptography In the Bounded Quantum-Storage Model* in *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science* (IEEE Computer Society, USA, 23rd Oct. 2005), 449–458. ISBN: 978-0-7695-2468-9. doi:10.1109/SFCS.2005.30.

[207] Yin, H.-L. & Chen, Z.-B. Finite-Key Analysis for Twin-Field Quantum Key Distribution with Composable Security. *Sci Rep* **9,** 17113. ISSN: 2045-2322. doi:10.1038/s41598-019-53435-4 (19th Nov. 2019).

[208] Kiesel, N., Schmid, C., Weber, U., Ursin, R. & Weinfurter, H. Linear Optics Controlled-Phase Gate Made Simple. *Phys. Rev. Lett.* **95,** 210505. doi:10.1103/PhysRevLett.95.210505 (18th Nov. 2005).

[209] Briegel, H. J., Browne, D. E., Dür, W., Raussendorf, R. & Van den Nest, M. Measurement-Based Quantum Computation. *Nature Phys* **5,** 19–26. ISSN: 1745-2481. doi:10.1038/nphys1157 (Jan. 2009).

[210] Englbrecht, M., Kraft, T. & Kraus, B. Transformations of Stabilizer States in Quantum Networks. *Quantum* **6,** 846. doi:10.22331/q-2022-10-25-846 (25th Oct. 2022).

[211] Van Rossum, G. & Drake, F. L. *Python 3 Reference Manual* ISBN: 1-4414-1269-7 (CreateSpace, Scotts Valley, CA, 2009).

[212] Harris, C. R. *et al.* Array Programming with NumPy. *Nature* **585,** 357–362. doi:10.1038/s41586-020-2649-2 (Sept. 2020).

[213] Hunter, J. D. Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering* **9,** 90–95. doi:10.1109/MCSE.2007.55 (2007).

[214] Tomamichel, M., Lim, C. C. W., Gisin, N. & Renner, R. Tight Finite-Key Analysis for Quantum Cryptography. *Nat Commun* **3,** 634. ISSN: 2041-1723. doi:10.1038/ncomms1631 (17th Jan. 2012).

[215] Grasselli, F., Kampermann, H. & Bruß, D. Finite-Key Effects in Multipartite Quantum Key Distribution Protocols. *New J. Phys.* **20,** 113014. ISSN: 1367-2630. doi:10.1088/1367-2630/aaec34 (Nov. 2018).

Appendices

A

Proof of Thm. 1

This appendix gives the proof of Thm. 1. It is directly sourced from Pub. **[F]** ([68]) where it is included as appendix *A*. However, the notation σ_x has been changed to X, etc. Still, the proof uses e.g. the notation $\sigma_i^{a_i}$ to represent any of the three Pauli operators X_i, Y_i or Z_i, indicated by $a_i \in \{x, y, z\}$[1].

For convenience, the theorem is first restated.

Theorem 3. (Pub. **[F]**) *No 2-island can have both a left and a right neighbour in V_G. This means that there is no node left of i or right of $i+1$ in V_G.*

Proof. The proof is by contradiction. First, fix a set V_G such that $\{i, i+1\} \subset V_G$ and let the post-measurement state $|\psi\rangle_{V_G}$ be locally equivalent to $|\mathrm{GHZ}_{V_G}\rangle$. Assume now that there are both $j < i$ and $k > i+1$ for which both $j, k \in V_G$. W.l.o.g. assume that j and k are the direct left- and right neighbour of i and $i+1$, respectively.

The generators for the linear cluster state are $\{l_{i_0} = Z_{i_-} X_{i_0} Z_{i_+}\}_{i_0 \in V_L}$. If the post-measurement state is locally equivalent to the GHZ state then there must exist a (reversible) generator transformation such that their support on i and $i+1$ coincides exactly with (the generators of) the GHZ state - up to local Clifford rotations. We will now show that, from a reversible transformation of the $\{l_{i_0}\}$, it is impossible to obtain such a set of generators when $j, i, i+1, k \in V_G$. This directly implies that a measurement pattern such that the GHZ state can be obtained is not possible.

(A set of) generators for the GHZ state are, $\{X_{V_G}\} \cup \{Z_{i_0} Z_{i_+}\}_{i_0 \in V_G}$, where it is implied that $Z_{i_+} = 1$ when $i_+ \notin V_G$. Focusing on i and $i+1$, the only generators with non-trivial support are $\{\alpha, \beta, \gamma, \delta\}$ being $\{\sigma_i^{a_i}, \sigma_i^{a_i}\sigma_{i+1}^{a_{i+1}}, \sigma_{i+1}^{a_{i+1}}, \sigma_i^{b_i}\sigma_{i+1}^{b_{i+1}}\}$, where $a_i, a_{i+1}, b_i, b_{i+1} \in \{x, y, z\}$ reflect the

[1] Note that the super-and subscripts have been interchanged w.r.t. the original presentation in Pub. **[F]**.

fact that the state is *locally equivalent* to the GHZ state. This implies that $a_i \neq b_i$ and $a_{i+1} \neq b_{i+1}$.

Similarly, only the generators l_{i-1}, l_i, l_{i+1} and l_{i+2} of the linear cluster state (i.e. those with support on i or $i+1$) can have a non-trivial contribution to the generator transformation on the vertices in question. Therefore, w.l.o.g., we can focus on just these four generators and restrict our attention to vertices i and $i+1$. If we show that there is no reversible transformation of $\{l_k\}_{k=\{i-1,i,i+1,i+2\}}$ to obtain $\{\alpha, \beta, \gamma, \delta\}$ when only considering these nodes, the theorem follows. We show there is no such transformation by exhaustive contradiction.

There are three different ways of creating generator α: **i)** $\alpha \propto l_{i-1} = Z_i$, **ii)** $\alpha \propto l_i \circ l_{i+2} = X_i$, **iii)** $\alpha \propto l_{i-1} \circ l_i \circ l_{i+2} = Y_i$, where '$\alpha \propto l_{i-1}$' should be read as '$l_{i-1}$ takes the role of α', and $\circ$ denotes the (qubit-wise) product (e.g. $l_i \circ l_{i+1} = X_i Z_{i+1} \circ Z_i X_{i+1} \hat{=} Y_i Y_{i+1}$, where the last equality is up to an irrelevant global phase). Similarly, there are three different ways of creating generator γ: **j)** $\gamma \propto l_{i+2} = Z_{i+2}$, **jj)** $\gamma \propto l_{i-1} \circ l_{i+1} = X_{i+2}$, **jjj** $\gamma \propto l_{i-1} \circ l_{i+1} \circ l_{i+2} = Y_{i+2}$. Picking e.g. **i)** and **j)** one sees that β is fixed at $\propto Z_i Z_{i+1}$. But this is $l_{i-1} \circ l_{i+2} \propto \alpha \circ \gamma$, which would not be a reversible transformation of the generators l_{i-1}, l_i, l_{i+1} and l_{i+2}. Any other pair from $\{$**i)**, **ii)**, **iii)**$\}$ and $\{$**j)**, **jj)**, **jjj)**$\}$ would also necessitate such a non-reversible transformation.

In essence, when viewing the generators as vectors over $\mathbb{F}_2^{2n}$ through the binary representation (see sec. 2.5 and [93]), the argument follows from the observation that (the vector associated with) β lies in the subspace spanned by (the vectors associated with) α and γ. As such there can never be a reversible (i.e. basis-preserving) operation on (the vectors associated with) l_{i-1}, l_i, l_{i+1} and l_{i+2} that obtains α, β and γ. □

B

Corrections for GHZ extraction in chapter 5

This appendix details the correction operators from chapter 5, that are necessary to obtain the target GHZ state from a linear cluster resource state.

The generators of the post-measurement state after measurement of the $|L_n\rangle$ state during maximal extraction are listed in **Tab.** 5.2. For easy reference, the generators are restated in **Tab.** B.1. As noted in chapter 5, these generators are related by a local Clifford operation to the generators of the GHZ state (see Def. 17).

	1	2	4	6	$\cdots$	$n-3$	$n-1$	n	ϕ
g_1	X	Z			$\ldots$				$+1$
g_3		Z	Z		$\ldots$				m_3
g_5			Z	Z	$\ldots$				m_5
$\ldots$					$\ldots$				$\ldots$
g_{n-2}					$\ldots$	Z	Z		m_{n-2}
g_n					$\ldots$		Z	X	$+1$
g_2'	Z	X	X		$\ldots$	X	X	Z	$+1$

Table B.1: After performing all measurements and removing the measured nodes, only those generators from **Tab.** 5.1 that commute with the measurement operators remain, which now carry the measurement outcomes $\{m_j = \pm 1\}$ as a phase. The post-measurement state is LC-equivalent to the target GHZ state (see Def. 17).

More specifically, a Hadamard operation on the first and last node realise the correct Pauli operators in the the generators with support on those qubits (i.e. g_1 and g_n become Z_1Z_2 and $Z_{n-1}Z_n$, respectively). Subsequently, the non-trivial phases of the generators (due to those measurements outcomes that were -1) can be removed by carefully applying a series of X operations to the qubits. Applying an X operator to any node j flips the phase of the generators g_{j-1} and g_{j+1}, so that there does not exist a one-to-one correspondence with the non-trivial phases of the generators and nodes to apply the X operators to.

In general, choosing what nodes to apply the X operator to amounts to solving the underdetermined system of linear equations

$$A\mathbf{x} = \mathbf{m}, \tag{B.1}$$

where $\mathbf{x}$ is the length-$|V_G|$ binary vector that encodes the choice of nodes to perform the X operations: if the k-th entry of $\mathbf{x}$ is 1, an X operator should be applied to the k-th node in V_G.

The vector $\mathbf{m} = [0, m_3, m_5, \ldots, m_{n-2}, 0]^T$ is a length-$(|V_G| - 1)$ vector containing the phases of the odd-indexed generators (note that the phases have been mapped back to binary, $\{+1, -1\} \to \{0, 1\}$), and the operator A is the $(|V_G| - 1) \times |V_G|$ matrix:

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 & \ldots & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & \ldots & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & \ldots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \ldots & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \ldots & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & \ldots & 0 & 0 & 1 & 1 \end{bmatrix}, \tag{B.2}$$

which has rank $|V_G| - 1$ and can be mapped to reduced echelon form by applying the matrix R that has zeros on its lower left triangle and ones on its diagonal and upper right triangle. This results in the matrix RA:

$$RA = \begin{bmatrix} 1 & 0 & 0 & \ldots & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & \ldots & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & \ldots & 0 & 0 & 0 & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & \ldots & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & \ldots & 0 & 0 & 1 & 1 \end{bmatrix}. \tag{B.3}$$

Combining (B.1) and (B.3) it follows that $RA\mathbf{x} = R\mathbf{m} = \mathbf{m_R}$ and thus that:

$$\mathbf{x} = [\mathbf{m_R}, 0]^T + \alpha[1, 1, \ldots, 1]^T, \tag{B.4}$$

where $\alpha = 0, 1$ is a free parameter reflecting the fact that A gives an underdetermined system. That two choices exist for the correction operator is not surprising following the fact that the vector $[1, 1, \ldots, 1]^T$ encodes the operator $X_1 X_2 \ldots X_n$, i.e. the generator g_2', which is part of the stabilizer of the GHZ state and can thus be applied to the state without non-trivially affecting it.

A jupyter notebook containing Python code to calculate what nodes to apply the X operation to is presented in Sup. **[sB]**, the supplementary material of Pub. **[F]**, which can be found at [127].

C

Details for fidelity estimation method of chapter 5

This appendix contains some details for the method to estimate the fidelity of the linear cluster state and GHZ state in the experimental realisation presented in chapter 5. In particular, it explains how to realise the n-qubit measurement operators as single-qubit measurements.

To estimate the fidelity, the expectation value $\mathbb{E}(P)$ for every $P \in \mathcal{S}_o$ or $P \in \mathcal{S}_e$ has to be determined, i.e. every element of the even and odd subgroup of the stabilizer (see (5.4)). All these elements are multi-qubit operators, so that measurements in their basis are hard to properly perform.

Indeed, measurements on the `IBMQ` devices are, as is often the case, restricted to single-qubit measurements. Thus, to properly measure e.g. the generator $Z_1Z_2 \in \mathcal{S}_o$ as a single operator would involve entangling gates to realise the desired multi-qubit operator from single-qubit operators. However, the measurement can be *simulated* by single-qubit measurements when one is only interested in the measurement outcome, instead of additionally its post-measurement state. More specifically, decomposing $Z_1Z_2 = \Pi_{+1} - \Pi_{-1}$ into its $+1$ and -1 eigenspace projectors, the term $\mathrm{tr}\left[\rho Z_1 Z_2\right]$ becomes:

$$\mathrm{tr}\left[\rho Z_1 Z_2\right] = \mathrm{tr}\left[\rho \Pi_{+1}\right] - \mathrm{tr}\left[\rho \Pi_{-1}\right]. \tag{C.1}$$

The eigenspace projectors Π_{+1} and Π_{-1} can be written in terms of the eigenspaces of the tensor factors of Z_1Z_2:

$$\begin{aligned} \Pi_{+1} =& \Pi_{+1}^{Z_1} \otimes \Pi_{+1}^{Z_2} + \Pi_{-1}^{Z_1} \otimes \Pi_{-1}^{Z_2}, \\ \Pi_{-1} =& \Pi_{+1}^{Z_1} \otimes \Pi_{-1}^{Z_2} + \Pi_{+1}^{Z_1} \otimes \Pi_{-1}^{Z_2}. \end{aligned} \tag{C.2}$$

Thus, measuring both qubits separately in the Z basis can reconstruct the

two-qubit measurement:

$$\begin{aligned}\mathrm{tr}\left[\rho Z_1 Z_2\right] = \\ &= \mathrm{tr}\left[\rho\left(\Pi_{+1}^{Z_1}\otimes\Pi_{+1}^{Z_2}\right)\right] + \mathrm{tr}\left[\rho\left(\Pi_{-1}^{Z_1}\otimes\Pi_{-1}^{Z_2}\right)\right] \\ &- \mathrm{tr}\left[\rho\left(\Pi_{+1}^{Z_1}\otimes\Pi_{-1}^{Z_2}\right)\right] - \mathrm{tr}\left[\rho\left(\Pi_{+1}^{Z_1}\otimes\Pi_{-1}^{Z_2}\right)\right].\end{aligned} \tag{C.3}$$

A single round of single-qubit Z measurements has the outcome $(\pm 1, \pm 1)$, which can be represented by the two-bit value $(0,0)$, $(0,1)$, $(1,0)$ or $(1,1)$. The terms in (C.3) are estimated by the relative frequencies of these outcomes in repeated measurements. If this measurement is repeated a total of n times, resulting in outcome statistics n_{00}, n_{01}, n_{10} and n_{11} (summing to n), the expectation value of the $Z_1 Z_2$ measurement is thus estimated by

$$\mathrm{tr}\left[\rho Z_1 Z_2\right] = \frac{n_{00} + n_{11} - n_{10} - n_{01}}{n}. \tag{C.4}$$

Any multi-qubit measurement operator that involves an identity operator can be reconstructed from the single-qubit measurement outcomes as well. More specifically, the $+1$ and -1 eigenspace projectors of operator $Z_1\mathbb{I}_2$ can again be decomposed in terms of the eigenspaces of the tensor factors of $Z_1 Z_2$. Because any state is the $+1$ eigenstate of the $\mathbb{I}$ operator, the decomposition now becomes:

$$\begin{aligned}\Pi_{+1} =& \Pi_{+1}^{Z_1}\otimes\Pi_{+1}^{Z_2} + \Pi_{+1}^{Z_1}\otimes\Pi_{-1}^{Z_2}, \\ \Pi_{-1} =& \Pi_{-1}^{Z_1}\otimes\Pi_{+1}^{Z_2} + \Pi_{-1}^{Z_1}\otimes\Pi_{-1}^{Z_2}.\end{aligned} \tag{C.5}$$

It follows, using the same methods as for the operator $Z_1 Z_2$, that:

$$\mathrm{tr}\left[\rho Z_1 \mathbb{I}_2\right] = \frac{n_{00} + n_{01} - n_{01} - n_{11}}{n}. \tag{C.6}$$

This method is straightforward to generalize, so it follows that the expectation value of any n-qubit operator containing only Z or $\mathbb{I}$ tensor factors, can be reconstructed from the measurement outcomes when every qubit is individually measured in the Z basis. Hence, the expectation value of all elements of the odd subgroup can be estimated using only one measurement setting.

The case for the even subgroup, where every operator consists of only X and $\mathbb{I}$ tensor factors, follows similarly when all qubits are individually measured in the X basis. Therefore, the fidelity can be estimated using only two measurement settings.

D

Subprotocols of ACKA

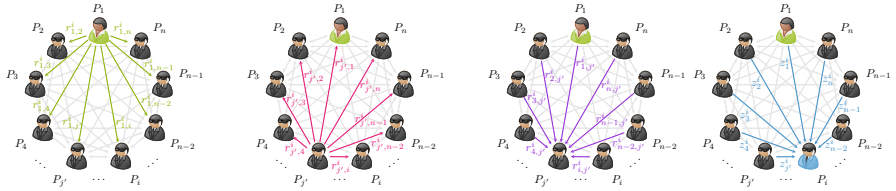
This chapter contains an overview of the subprotocols used in ACKA from chapter 8, i.e. Protocol I. The figures, including their captions, are directly sourced from Pub. **[A]**.

There are three different sub-protocols, that are all discussed in their own section. Specifically, NOTIFICATION is discussed in sec. D.1, AME is discussed in sec. D.2 and VERIFICATION is discussed in sec. D.3.

D.1 | NOTIFICATION

The first subprotocol, NOTIFICATION, allows Alice to anonymously notify every participant $\mathcal{B}_i$ that they are a participant, where the participants $\mathbf{P}$ is a subset of the network of het choosing. It is based on the NOTIFICATION protocol presented in [205], but slightly adapted. The protocol is entirely classical, in the sense that no quantum communication is used, and it results in every participant $\mathcal{B}_i$ knowing that they are a participant, while anyone from $\bar{\mathbf{P}}$ does not learn anything.

Fig. D.1, originally presented in appendix B of Pub. **[A]** ([2]) can be helpful to understand the protocol. It details the public communication of one round (i.e. for one fixed i) of the protocol, and shows how the important information is encoded into the parities of all this communication.



(A) Step item 1a with $j_a = 1$. (B) Step 1b with $j = j'$. (C) Step 2 with $k = j'$ (D) Step 3 with $i = i_b$.

j \ k	1	2	3	⋯	j'	⋯	i	⋯	n
1	$r^i_{1,1}$	$r^i_{1,2}$	$r^i_{1,3}$	⋯	$r^i_{1,j'}$	⋯	$r^i_{1,i}$	⋯	$r^i_{1,n}$
2	$r^i_{2,1}$	$r^i_{2,2}$	$r^i_{2,3}$	⋯	$r^i_{2,j'}$	⋯	$r^i_{2,i}$	⋯	$r^i_{2,n}$
3	$r^i_{3,1}$	$r^i_{3,2}$	$r^i_{3,3}$	⋯	$r^i_{3,j'}$	⋯	$r^i_{3,i}$	⋯	$r^i_{3,n}$
⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯
j'	$r^i_{j',1}$	$r^i_{j',2}$	$r^i_{j',3}$	⋯	$r^i_{j',j'}$	⋯	$r^i_{j',i}$	⋯	$r^i_{j',n}$
⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯
i	$r^i_{i,1}$	$r^i_{i,2}$	$r^i_{i,3}$	⋯	$r^i_{i,j'}$	⋯	$r^i_{i,i}$	⋯	$r^i_{i,n}$
⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯
n	$r^i_{n,1}$	$r^i_{n,2}$	$r^i_{n,3}$	⋯	$r^i_{n,j'}$	⋯	$r^i_{n,i}$	⋯	$r^i_{n,n}$
$\bigoplus_j r^i_{j,k}$	z^i_1	z^i_2	z^i_3	⋯	$z^i_{j'}$	⋯	z^i_i	⋯	z^i_n

FIGURE D.1: Visualization of subprotocol 1. The table contains all $r^i_{j,k}$ for a fixed node $P_i \in \mathcal{N}$ in the NOTIFICATION protocol. Here, we identify Alice with P_1. She chooses $\{r^i_{1,k}\}_{k=1}^n$ and sends them to P_k in Step 1a (Fig. D.1a). Note that only if P_i is a receiver, the green row adds up to 1 (mod 2); otherwise to 0 (mod 2). Analogously, the pink highlighting shows Step 1b from the perspective of $P_{j'}$ (Fig. D.1b). This and all other rows add up to 0 (mod 2). The $\{r^i_{j,j'}\}_{j=1}^n$ that $P_{j'}$ receives in Step 2 (Fig. D.1c) are highlighted in purple. The last row, highlighted in blue, shows the $\{z^i_k\}_{k=1}^n$ received by P_i in Step 3 (Fig. D.1d). By construction, only if P_i is a receiver, it adds up to 1 (mod 2).

Protocol V - NOTIFICATION	
Input:	Alice's choice of m receivers.
Goal:	The m receivers get notified.

For agent $i = 1, \ldots, n$:

1. All agents $j \in \{1, \ldots, n\}$ do the following.
   (a) When j corresponds to Alice (j_a), and i is not a receiver, she chooses n random bits $\{r^i_{j,k}\}^n_{k=1}$ such that $\bigoplus^n_{k=1} r^i_{j,k} = 0$. If i is a receiver, she chooses n random bits such that $\bigoplus^n_{k=1} r^i_{j,k} = 1$. She sends bit $r^i_{j,k}$ to agent k.
   (b) When $j \neq j_a$, the agent chooses n random bits $\{r^i_{j,k}\}^n_{k=1}$ such that $\bigoplus^n_{k=1} r^i_{j,k} = 0$ and sends bit $r^i_{j,k}$ to agent k.
2. All agents $k \in \{1, \ldots, n\}$ receive $\{r^i_{j,k}\}^n_{j=1}$, compute $z^i_k = \bigoplus^n_{j=1} r^i_{j,k}$ and send it to agent i.
3. Agent i takes the received $\{z^i_k\}^n_{k=1}$ to compute $z^i = \bigoplus^n_{k=1} z^i_k$; if $z^i = 1$ they are thereby notified to be a designated receiver.

Analysis: The correctness of the protocol follows from the analysis in [2, 205]. The anonymity of the protocol works because of the fact that all important information (i.e. the bit z^i encoding the role of node i) is encoded into the *parity* of all public communication. Therefore, at any moment during the protocol, an adversary can only reveal the role of node i by corrupting all participants in the network, including $\mathcal{A}$ and node i themselves.

D.2 | ANONYMOUS MULTIPARTITE ENTANGLEMENT

The next subprotocol, ANONYMOUS MULTIPARTITE ENTANGLEMENT (AME), allows the participants $\mathbf{P}$ to extract a $|\mathrm{GHZ}_{m+1}\rangle$ state on just their qubits from the $|\mathrm{GHZ}_{\mathcal{N}}\rangle$ state on the entire network. It effectively removes the non-participants $\bar{\mathbf{P}}$ from the network state, while keeping the identity of the participants $\mathbf{P}$ hidden. **Fig.** D.2 contains a visualization of the steps of the protocol.

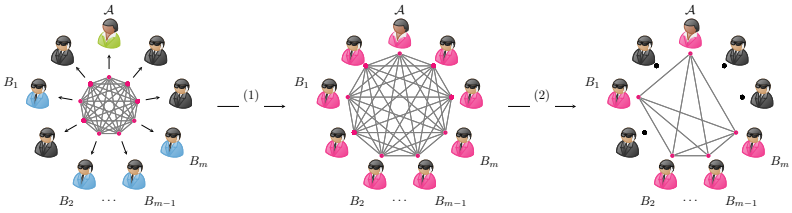


Figure D.2: Visualization of AME. First, a $|\mathrm{GHZ}_n\rangle$ state gets distributed between all nodes of the network. Even though the participants secretly play a special role, after distribution (step (1)) they are indistinguishable from all other nodes in the network. At step (2) all non-participants $\bar{\mathbf{P}}$ measure their qubit in the X basis, but the Bobs do nothing; after a correction by Alice the state of the network is $|\mathrm{GHZ}_{m+1}\rangle$ for the participants, disentangled from all other nodes in the network.

Protocol VI - ANONYMOUS MULTIPARTITE ENTANGLEMENT	
Input:	$\|\mathrm{GHZ}_n\rangle$ state; Alice knowing $\mathbf{P}$ and $\bar{\mathbf{P}}$
Goal:	A $\|\mathrm{GHZ}_{m+1}\rangle$ state shared between $\mathbf{P}$.

1: Alice and the Bobs each draw a random bit x_i. Everyone else measures in the X basis, yielding a measurement outcome bit x_j for $j \in \bar{\mathbf{P}}$.

2: All parties broadcast their bits in a random order or, if possible, simultaneously.

3: Alice applies a Z gate if the parity of the non-participating parties' bits is odd, i.e. if and only if $\bigoplus_{j\in\bar{\mathbf{P}}} x_j = 1$.

Analysis: The correctness of the protocol follows from the analysis in [2, 26]. Since Alice has chosen herself who in the network belongs to $\mathbf{P}$, she also knows what announced bits belong to the non-participants $\bar{\mathbf{P}}$. Therefore, she can make the last step of the subprotocol without any ambiguity. The fact that

the resulting state for the participants $\mathbf{P}$ is indeed the desired state $|\mathrm{GHZ}_{\mathbf{P}}\rangle$ follows from a careful rewriting of the GHZ state. First, the measurements of the non-participants in the X basis are the same as measurements in the Z basis preceded by a Hadamard operation on those qubits.

It helps to first investigate how a single Hadamard operation applied on the last qubit affects the state:

$$\begin{aligned} & H_n \left|\mathrm{GHZ}_n\right\rangle \\ =& \frac{1}{\sqrt{2}} \left(|0,\ldots,0,+\rangle_{1,\ldots,n-1,n} + |1,\ldots,1,-\rangle_{1,\ldots,n-1,n}\right) \\ =& \frac{1}{2} |0,\ldots,0\rangle_{1,\ldots,n-1} \otimes (|0\rangle + |1\rangle)_n \\ +& \frac{1}{2} |1,\ldots,1\rangle_{1,\ldots,n-1} \otimes (|0\rangle - |1\rangle)_n \\ =& \frac{1}{\sqrt{2}} \left(|\mathrm{GHZ}_{1,\ldots,n-1}\rangle \otimes |0\rangle_n + Z_A |\mathrm{GHZ}_{1,\ldots,n-1}\rangle \otimes |1\rangle_n\right). \end{aligned} \tag{D.1}$$

Thus, if the node in the network with the last qubit applies a Hadamard operation and subsequently measures in the Z basis, resulting in outcome 0 or 1, the state of the rest of the network becomes either $|\mathrm{GHZ}_{1,\ldots,n-1}\rangle$ or $Z_A |\mathrm{GHZ}_{1,\ldots,n-1}\rangle = \frac{1}{\sqrt{2}} (|0,\ldots,0\rangle - |1,\ldots,1\rangle)$, respectively[1].

A similar analysis for the state $Z_A |\mathrm{GHZ}_{1,\ldots,n-1}\rangle \otimes |1\rangle_n$ is straightforward, so that the general case follows readily. Applying a Hadamard operation on all qubits of the non-participants $\bar{\mathbf{P}}$ results in the state

$$\sum_{x\in\{0,1\}^{|\bar{\mathbf{P}}|}|\text{even 1}} |x\rangle_{\bar{\mathbf{P}}} \otimes |\mathrm{GHZ}_{\mathbf{P}}\rangle + \sum_{x\in\{0,1\}^{|\bar{\mathbf{P}}|}|\text{odd 1}} |x\rangle_{\bar{\mathbf{P}}} \otimes Z_A |\mathrm{GHZ}_{\mathbf{P}}\rangle . \tag{D.2}$$

Thus, if the non-participants announce a set of measurement outcomes $\{x_i\}_{i\in\bar{\mathbf{P}}}$ s.t. $\bigoplus_{i\in\bar{\mathbf{P}}} x_i = 0$, it means that the state of the participants $\mathbf{P}$ is $|\mathrm{GHZ}_{\mathbf{P}}\rangle$. Similarly, if the non-participants announce a set of measurement outcomes $\{x_i\}_{i\in\bar{\mathbf{P}}}$ s.t. $\bigoplus_{i\in\bar{\mathbf{P}}} x_i = 1$, it means that the state of the participants $\mathbf{P}$ is $Z_A |\mathrm{GHZ}_{\mathbf{P}}\rangle$. A final Z_A operation by Alice in the latter case results in the desired state.

All participants announce a random bit to hide their identity.

[1] Note that it has been implicitly assumed that $\mathcal{A}$ is not the last node.

D.3 | VERIFICATION

Although the state for the participants after subprotocol 2 *should* be the $|\text{GHZ}_{\mathbf{P}}\rangle$ state, any non-participant can arbitrarily deviate from the protocol. In particular, a non-participant $[u] \in \bar{\mathbf{P}}$ could not perform the measurement dictated by subprotocol 2, but instead announce a bit $x_u = 0$. The state of the network would then be $|\text{GHZ}_{m+2}\rangle_{\mathbf{P}+[u]}$, which means that they could participate in the key generation - thereby learning the key completely undetected and thus compromising security. It is therefore vital that the network state is *verified* - if the state of the participants $\mathbf{P}$ is indeed the GHZ state, then by the monogamy of entanglement the state of rest of the network is completely separable from the participants. The third subprotocol, `VERIFICATION`, allows Alice to perform this verification, so that she can `ACCEPT` or `REJECT` the state.

`Protocol VII - VERIFICATION`

`Input:` A shared state between $|\mathbf{P}| = m + 1$ parties.
`Goal:` Alice `ACCEPTS` or `REJECTS` the shared state as $|\text{GHZ}_{m+1}\rangle$.

---

1. Every B_i draws a random bit b_i and measures in the X or Y basis if it equals 0 or 1 respectively, obtaining a measurement outcome o_i.
2. Everyone broadcasts (b_i, o_i), including Alice, who chooses her bits (b_0, o_0) at random.
3. Alice resets her bit such that $\sum_{i=0}^{m} b_i = 0 \pmod 2$. She measures in the X or Y basis if her bit equals 0 or 1 respectively, thereby additionally resetting o_0.
4. If and only if $\frac{1}{2}\sum_i b_i + \sum_{i=0}^{m} o_i = 0 \pmod 2$, Alice `ACCEPTS` the state, otherwise she `REJECTS`.

Analysis: The protocol is inspired by that of [200] and largely follows it, although adapted for the specific desired output state $|\text{GHZ}_{m+1}\rangle$. This is a stabilizer state, and during the protocol the participants essentially measure an element of the stabilizer $\mathcal{S}_{\text{GHZ}}$, determining if indeed the network state is in the correct eigenspace.

Indeed, the choice of measurement basis that Alice makes in item 3 guarantees that the collective measurement operator of all participants contains an even number of Y operators, where the rest of the qubits are X operators. Thus, the collective measurement operator is an element of $\mathcal{S}_{\text{GHZ}}$ that at least 'uses' the generator $g_{m+1} = X_0 \otimes X_1 \otimes, \ldots, \otimes X_m$ (see Def. 17), and then any number of the other generators $\{g_i\}_{i\in\{,1,\ldots,m\}}$.

Similar to the method described in chapter C, these stabilizer elements themselves are not actually measured, but they are reconstructed from the single-qubit measurements by every participant.

Consider e.g. the operator $P = X_{\mathbf{P}} = X_0 \otimes X_1 \otimes \cdots \otimes X_m$, the stabilizer element associated with every participant measuring in the X basis (i.e. $b_i = 0$ for all $i \in \mathbf{P}$). The state must be in the $+1$-eigenspace of this operator, whose projector Π^P_{+1} can be written in terms of the projectors $\Pi^{X_i}_{+1}$ of the single-qubit measurement operators.

The single-qubit measurements result in outcomes o_i; the outcomes o_i are all either 0 or 1, for the $+1$ or -1 eigenspace of the measurement operator X_j, respectively. These eigenspaces have projectors which can be written as $\Pi^X_{o_j}$, so that all measurements combined have a projector $\Pi^X_{o_0} \otimes \Pi^X_{o_1} \otimes \cdots \otimes \Pi^X_{o_m}$. The desired stabilizer element projection operator Π^P_{+1} is exactly the sum of all such measurements with an even number of -1 eigenspaces $\Pi^X_{o_j}$:

$$\Pi^P_{+1} = \sum_{o \in \{0,1\}^{|\mathbf{P}|} | \text{even } 1} \Pi^X_{o_0} \otimes \Pi^X_{o_1} \otimes \cdots \otimes \Pi^X_{o_m} \tag{D.3}$$

The condition $\sum_{i=0}^m o_i = 0$ upon which Alice ACCEPTS ensures that there are an even number of -1 eigenspaces. Since any such set of outcomes is accepted, the condition thus ensures that the state can not be in any other eigenspace, and thus must be in the correct eigenspace of the stabilizer element.

If e.g. $b_0 = b_1 = 1$ and all other b_i's are 0, the simulated stabilizer element is $Y_0 \otimes Y_1 \otimes X_2 \otimes \cdots \otimes X_m$. Technically, this is not an element of the stabilizer, but rather $-YYX \ldots X \in \mathcal{S}_{\text{GHZ}}$. Thus, the $+1$ and -1 eigenspaces of this operator are swapped, so that instead of an *even* number of 1's, there should be an *odd* number of 1's in the measurement outcomes o_i. This occurs any time there is a total number of Y operators which is two times an odd number - the condition on which Alice thus should accept is $\frac{1}{2}\sum_i b_i + \sum_{i=0}^m o_i = 0 \pmod 2$.

Due to the nature of the protocol, the choice of stabilizer element that is tested is random; from the perspective of an adversary the state is tested by a random, unknown stabilizer element. In [200] it is shown that any state that passes this random check must be exceedingly close to the GHZ state. It is thus of vital importance that the choice of measurement bases (i.e. the b_i) are announced at the same time as the measurement outcomes o_i - this ensures that the measurement are random and unknown to Eve until *after* the measurements have taken place, to ensure security.

A final small technical point is raised by the fact that not *every* stabilizer element is always chosen. The protocol can only realize those elements that always 'use' the last generator $X_0 \otimes X_1 \otimes \cdots \otimes X_m$, so that only half of all 2^n elements in the stabilizer can be tested. However, the shared $+1$ eigenspace

of all these operators is still a unique state, exactly the desired GHZ state[2].

[2]Remember that just n generators are enough to specify a stabilizer state as the unique state in the shared $+1$ eigenspace of the generators - so the set of 2^{n-1} stabilizer elements is, in a way, too much.

E

Anonymity in ACKA

This appendix details the anonymity of ACKA, i.e. Protocol I. This appendix is directly sourced from Pub. **[A]** ([2]) where it is included as appendix B.

Anonymity is proven in terms of Def. 31, which is restated first.

Definition 33. *Let* $\mathbf{P} \subset \mathcal{N}$ *be the arbitrarily-sized set of participants of an anonymous protocol and Eve be an adversary that wishes to learn* $\mathbf{P}$*. Furthermore, let* $\mathcal{I}_{\text{Eve}}$ *be the information regarding* $\mathbf{P}$ *that Eve has both beforehand or trivially learns by corrupting any number of non-participants. Then, the protocol is* anonymous *if for any subset* $\mathbf{G} \subset \mathcal{N}$

$$\Pr\left(\mathbf{G} = \mathbf{P} | \mathcal{I}^{+}_{\text{Eve}}, \mathcal{I}_{\text{Eve}}\right) = \Pr\left(\mathbf{G} = \mathbf{P} | \mathcal{I}_{\text{Eve}}\right), \tag{E.1}$$

where $\mathcal{I}^{+}_{\text{Eve}}$ *is the information that Eve additionally learns during the protocol, which includes all public communication and any quantum systems she has access to.*

In order to satisfy Def. 33, $\mathcal{I}^{+}_{\text{Eve}}$ should not change Eve's probability distribution of uncovering the partitioning of $\mathcal{N}$ into its constituents; it does not reveal anything about $\mathbf{P}$, $\mathbf{H}$ or – implicitly – about $\mathbf{C}$. Apart from the trivial attacker $\mathcal{A}$ we consider three different types of adversary Eve, namely any other party in $\mathbf{P}$, any party in $\mathbf{H}$ or all parties in $\mathbf{C}$. The symbols ★, ✓ and ⊸ are used for the AME subprotocol, the **Verification** round and the **Keygen** round, respectively.

We use the structure of **Tab.** E.1 to prove anonymity with respect to all different types of Eve. AME and VERIFICATION will be examined in sec. E.1 and sec. E.2, respectively. The **Keygen** rounds do not require any public communication and will be examined in sec. E.3. To prove our claim we consider the following two aspects. The *public communication* (cf. **Tab.** E.2) throughout the protocol does not help Eve to reveal the roles of the participating parties.

Eve \ role	A	$B_i \in \mathbf{P} \setminus A$	$P_j \in \mathbf{H}$	$P_k \in \mathbf{C}$
A	trivial	trivial	irrelevant	irrelevant
$B_i \in \mathbf{P} \setminus A$	$\bigstar_3$ $\checkmark_2$ $\rightsquigarrow_3$	$\bigstar_3$ $\checkmark_2$ $\rightsquigarrow_3$	$\bigstar_1$ $\checkmark_2$ $\rightsquigarrow_1$	$\bigstar_3$ $\checkmark_2$ $\rightsquigarrow_1$
$P_j \in \mathbf{H}$	$\bigstar_2$ $\checkmark_1$ $\rightsquigarrow_1$	$\bigstar_2$ $\checkmark_1$ $\rightsquigarrow_1$	$\bigstar_2$ $\checkmark_1$ $\rightsquigarrow_2$	$\bigstar_2$ $\checkmark_1$ $\rightsquigarrow_1$
$P_k \in \mathbf{C}$	$\bigstar_3$ $\checkmark_3$ $\rightsquigarrow_1$	$\bigstar_3$ $\checkmark_3$ $\rightsquigarrow_1$	$\bigstar_1$ $\checkmark_3$ $\rightsquigarrow_1$	trivial

Table E.1: The rows are labelled by the types of adversary and the columns by the roles that Eve may try to uncover. The first row is mostly trivial, since the protocol is designed such that $\mathcal{A}$ chooses the partitioning $\mathcal{N} = \mathbf{P} \cup \bar{\mathbf{P}}$ herself and it is irrelevant that she is unaware of who in $\bar{\mathbf{P}}$ is colluding. The arguments corresponding to the symbols are given in sec. E.1, sec. E.2 and sec. E.3. Note that Alice is referred to as A instead of $\mathcal{A}$.

We prove this by showing that all public communication is indistinguishable from Eve's point of view. As $\mathcal{A}$ announces only uniformly random and uncorrelated bits, we will show the same for the parties in $\mathbf{P} \setminus \mathcal{A}$, $\mathbf{H}$ and $\mathbf{C}$ from any Eve's perspective. Likewise, the *quantum states* accessible to Eve do not help her to reveal the roles of the participating parties, even given access to the public communication. This means that the post-measurement states of Eve can neither be correlated with the measurement outcomes of other parties, nor with any direct information regarding their roles. Note that the global quantum state may encode such information regarding the roles as long as it is not accessible to anyone but Alice.

E.1 | Anonymity during AME

At the start of AME, the shared quantum state is as given by the following equation:

$$|\mathcal{N}\rangle \approx_\epsilon \frac{1}{\sqrt{2}} \left(|0 \dots 0\rangle_{\mathbf{H}} \otimes |\Psi\rangle_{\mathbf{C}} + |1 \dots 1\rangle_{\mathbf{H}} \otimes |\Phi\rangle_{\mathbf{C}} \right), \tag{E.2}$$

for some arbitrary states $|\Psi\rangle_{\mathbf{C}}$ and $|\Phi\rangle_{\mathbf{C}}$ held by the corrupted parties $\mathbf{C}$. While AME prescribes measurements to both $\mathbf{H}$ and $\mathbf{C}$, the parties in $\mathbf{C}$ might not measure and announce something unrelated to their arbitrary actions on the quantum state – therefore we now only calculate the probability of the measurement outcomes $\mu^\alpha_{\mathbf{H}} = \{\mu_j \mid j \in \mathbf{H}\}$ of $\mathbf{H}$ taking values $x^\alpha_{\mathbf{H}} = \{x^\alpha_i\} \in \{0,1\}^{|\mathbf{H}|}$. We want to show that they are uniformly random and that there are no correlations between the outcomes and any Eve that she might exploit, where Eve might be anyone in the network but Alice. That is, we want to

	AME	Verification
$\mathcal{A}$	random bit r_0	random bits (b_0, o_0)
$B_i \in \mathbf{P} \setminus \mathcal{A}$	random bit r_i	random bit b_i, outcome bit o_i
$P_j \in \mathbf{H}$	outcome bit x_j	random bits (b_j, o_j)
$P_k \in \mathbf{C}$	arbitrary bit $\tilde{x}_k$	arbitrary bits $(\tilde{b}_k, \tilde{o}_k)$

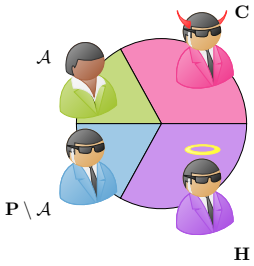


Table E.2: Overview of all public communication for any party in $\mathcal{N} := \mathbf{P} \cup \mathbf{H} \cup \mathbf{C}$ when running AME and VERIFICATION. The communication summarized in the two columns needs to be indistinguishable from the perspective of any Eve. Since $\mathcal{A}$ only announces uniformly random and uncorrelated bits, all other communication must follow the same probability distribution. Only the communication from $\mathbf{C}$ can in principle diverge – should they choose not to hide their identities.

show

$$\Pr\left(\mu^{\alpha}_{\mathbf{H}} = x^{\alpha}_{\mathbf{H}} \mid \mathcal{I}^{+}_{\mathrm{Eve}}, \mathcal{I}_{\mathrm{Eve}}\right) = \Pr\left(\mu^{\alpha}_{\mathbf{H}} = x^{\alpha}_{\mathbf{H}}\right) = \frac{1}{2^{|\mathbf{H}|}}, \tag{E.3}$$

where the second equality implies that the probability distribution of the measurement outcomes is uniform and the first equality implies that there are no correlations between the information accessible to Eve – including her quantum state – and the measurement outcomes. Moreover, we also want to show that the post-measurement state does not possess any other correlations regarding the roles of the parties that are accessible or exploitable by Eve.

The measurements on $\mathbf{H}$ in AME are a PVM with outcomes $\{x^{\alpha}_{\mathbf{H}}\}$ and associated projectors:

$$X^{\alpha}_{\mathbf{H}} := H_{\mathbf{H}} \left|x^{\alpha}_{\mathbf{H}}\right\rangle\!\left\langle x^{\alpha}_{\mathbf{H}}\right|_{\mathbf{H}} H_{\mathbf{H}} = \bigotimes_{j \in \mathbf{H}} H_j \left|x^{\alpha}_{j}\right\rangle\!\left\langle x^{\alpha}_{j}\right|_{j} H_j, \tag{E.4}$$

which results in the probability of the measurement outcome $\mu^{\alpha}_{\mathbf{H}}$ taking the

value $x^{\alpha}_{\mathbf{H}}$ being given by

$$
\begin{aligned}
\Pr(\mu^{\alpha}_{\mathbf{H}} = x^{\alpha}_{\mathbf{H}}) &= \operatorname{tr}\left[X^{\alpha}_{\mathbf{H}} \left|\mathcal{N}\right\rangle\!\left\langle\mathcal{N}\right|\right] \\
&= \frac{1}{2}\operatorname{tr}\left[\left(\left|0\ldots0\right\rangle\!\left\langle 0\ldots0\right|_{\mathbf{P}}\right)\right]\operatorname{tr}\left[X^{\alpha}_{\mathbf{H}}\left|0\ldots0\right\rangle\!\left\langle 0\ldots0\right|_{\mathbf{H}}\right]\operatorname{tr}\left[\left|\Psi\right\rangle\!\left\langle\Psi\right|_{\mathbf{C}}\right] \\
&+ \frac{1}{2}\operatorname{tr}\left[\left(\left|0\ldots0\right\rangle\!\left\langle 1\ldots1\right|_{\mathbf{P}}\right)\right]\operatorname{tr}\left[X^{\alpha}_{\mathbf{H}}\left|0\ldots0\right\rangle\!\left\langle 1\ldots1\right|_{\mathbf{H}}\right]\operatorname{tr}\left[\left|\Psi\right\rangle\!\left\langle\Phi\right|_{\mathbf{C}}\right] \\
&+ \frac{1}{2}\operatorname{tr}\left[\left(\left|1\ldots1\right\rangle\!\left\langle 0\ldots0\right|_{\mathbf{P}}\right)\right]\operatorname{tr}\left[X^{\alpha}_{\mathbf{H}}\left|1\ldots1\right\rangle\!\left\langle 0\ldots0\right|_{\mathbf{H}}\right]\operatorname{tr}\left[\left|\Phi\right\rangle\!\left\langle\Psi\right|_{\mathbf{C}}\right] \\
&+ \frac{1}{2}\operatorname{tr}\left[\left(\left|1\ldots1\right\rangle\!\left\langle 1\ldots1\right|_{\mathbf{P}}\right)\right]\operatorname{tr}\left[X^{\alpha}_{\mathbf{H}}\left|1\ldots1\right\rangle\!\left\langle 1\ldots1\right|_{\mathbf{H}}\right]\operatorname{tr}\left[\left|\Phi\right\rangle\!\left\langle\Phi\right|_{\mathbf{C}}\right] \\
&= \frac{1}{2}\operatorname{tr}\left[\left(\bigotimes_{j\in\mathbf{H}} H_j \left|x^{\alpha}_j\right\rangle\!\left\langle x^{\alpha}_j\right|_j H_j\right)\left|0\ldots0\right\rangle\!\left\langle 0\ldots0\right|_{\mathbf{H}}\right] \\
&+ \frac{1}{2}\operatorname{tr}\left[\left(\bigotimes_{j\in\mathbf{H}} H_j \left|x^{\alpha}_j\right\rangle\!\left\langle x^{\alpha}_j\right|_j H_j\right)\left|1\ldots1\right\rangle\!\left\langle 1\ldots1\right|_{\mathbf{H}}\right] \\
&= \frac{1}{2}\left(\prod_{i\in\mathbf{H}}\left|\langle x^{\alpha}_i|+\rangle\right|^2 + \prod_{i\in\mathbf{H}}\left|\langle x^{\alpha}_i|-\rangle\right|^2\right) \\
&= \frac{1}{2}\left(\frac{1}{2^{|\mathbf{H}|}} + \frac{1}{2^{|\mathbf{H}|}}\right) = \frac{1}{2^{|\mathbf{H}|}}.
\end{aligned}
\tag{E.5}
$$

This satisfies the second equality in Eq. (E.3), showing that the measurement outcomes are uniformly random, thereby ensuring that all the communication of the AME column of Tab. E.2 is indistinguishable – excluding the trivial case where $\mathbf{C}$ reveals itself. The global post-measurement state $\rho_{\text{post}\texttt{AME}}$

is then

$$\begin{aligned}
\rho_{\texttt{postAME}} &= X^{\alpha}_{\mathbf{H}} \left|\mathcal{N}\right\rangle\!\left\langle\mathcal{N}\right| X^{\alpha}_{\mathbf{H}} \\
&= \frac{1}{2}\left(\left|0\ldots0\right\rangle\!\left\langle 0\ldots0\right|_{\mathbf{P}}\right) \otimes X^{\alpha}_{\mathbf{H}} \left|0\ldots0\right\rangle\!\left\langle 0\ldots0\right|_{\mathbf{H}} X^{\alpha}_{\mathbf{H}} \otimes \left|\Psi\right\rangle\!\left\langle\Psi\right|_{\mathbf{C}} \\
&+ \frac{1}{2}\left(\left|0\ldots0\right\rangle\!\left\langle 1\ldots1\right|_{\mathbf{P}}\right) \otimes X^{\alpha}_{\mathbf{H}} \left|0\ldots0\right\rangle\!\left\langle 1\ldots1\right|_{\mathbf{H}} X^{\alpha}_{\mathbf{H}} \otimes \left|\Psi\right\rangle\!\left\langle\Phi\right|_{\mathbf{C}} \\
&+ \frac{1}{2}\left(\left|1\ldots1\right\rangle\!\left\langle 0\ldots0\right|_{\mathbf{P}}\right) \otimes X^{\alpha}_{\mathbf{H}} \left|1\ldots1\right\rangle\!\left\langle 0\ldots0\right|_{\mathbf{H}} X^{\alpha}_{\mathbf{H}} \otimes \left|\Phi\right\rangle\!\left\langle\Psi\right|_{\mathbf{C}} \\
&+ \frac{1}{2}\left(\left|1\ldots1\right\rangle\!\left\langle 1\ldots1\right|_{\mathbf{P}}\right) \otimes X^{\alpha}_{\mathbf{H}} \left|1\ldots1\right\rangle\!\left\langle 1\ldots1\right|_{\mathbf{H}} X^{\alpha}_{\mathbf{H}} \otimes \left|\Phi\right\rangle\!\left\langle\Phi\right|_{\mathbf{C}} \\
&= \frac{1}{2}\left(\left|0\ldots0\right\rangle\!\left\langle 0\ldots0\right|_{\mathbf{P}}\right) \otimes \left|\mathbf{H}\right\rangle\!\left\langle\mathbf{H}\right| \otimes \left|\Psi\right\rangle\!\left\langle\Psi\right|_{\mathbf{C}} \\
&+ \frac{1}{2}\left(\left|0\ldots0\right\rangle\!\left\langle 1\ldots1\right|_{\mathbf{P}}\right) \otimes (-1)^{\Delta(x^{\alpha}_{\mathbf{H}})} \left|\mathbf{H}\right\rangle\!\left\langle\mathbf{H}\right| \otimes \left|\Psi\right\rangle\!\left\langle\Phi\right|_{\mathbf{C}} \\
&+ \frac{1}{2}\left(\left|1\ldots1\right\rangle\!\left\langle 0\ldots0\right|_{\mathbf{P}}\right) \otimes (-1)^{\Delta(x^{\alpha}_{\mathbf{H}})} \left|\mathbf{H}\right\rangle\!\left\langle\mathbf{H}\right| \otimes \left|\Phi\right\rangle\!\left\langle\Psi\right|_{\mathbf{C}} \\
&+ \frac{1}{2}\left(\left|1\ldots1\right\rangle\!\left\langle 1\ldots1\right|_{\mathbf{P}}\right) \otimes \left|\mathbf{H}\right\rangle\!\left\langle\mathbf{H}\right| \otimes \left|\Phi\right\rangle\!\left\langle\Phi\right|_{\mathbf{C}} \\
&= \left|\mathcal{N}_{\texttt{postAME}}\right\rangle\!\left\langle\mathcal{N}_{\texttt{postAME}}\right|,
\end{aligned} \tag{E.6}$$

where $\left|\mathcal{N}_{\texttt{postAME}}\right\rangle\!\left\langle\mathcal{N}_{\texttt{postAME}}\right|$ is the pure state

$$\left|\mathcal{N}_{\texttt{postAME}}\right\rangle = \frac{1}{\sqrt{2}}\left(\left|0\ldots0\right\rangle_{\mathbf{P}} \otimes \left|\Psi\right\rangle_{\mathbf{C}} + (-1)^{\Delta(x^{\alpha}_{\mathbf{H}})} \left|1\ldots1\right\rangle_{\mathbf{P}} \otimes \left|\Phi\right\rangle_{\mathbf{C}}\right) \otimes \left|\mathbf{H}\right\rangle, \tag{E.7}$$

showing that the only correlation between the measurement outcome and the state on $\mathbf{P} \cup \mathbf{C}$ is in the phase, where one could in principle learn the parity of the measurement outcome $x^{\alpha}_{\mathbf{H}}$. However, any such phase estimation is impossible if one does not have access to the complete state (i.e. tracing out $\mathbf{P}$ that do not collude with Eve results in a state on $\mathbf{C}$ that is uncorrelated with the measurement outcome $x^{\alpha}_{\mathbf{H}}$). This means that the post-measurement state of any attacker in $\mathbf{P} \setminus \mathcal{A}$ or $\mathbf{C}$ is uncorrelated from the measurement outcome $x^{\alpha}_{\mathbf{H}}$ and the roles of $\mathbf{H}$. Therefore, for either of these types of Eve everyone in $\mathbf{H}$ remains anonymous (cf. $\bigstar_1$ in **Tab.** E.1).

Furthermore $\mathbf{H}$ is disentangled from the rest of the network and $\left|\mathbf{H}\right\rangle$ itself is separable over the constituents of $\mathbf{H}$. Therefore, nobody in $\mathbf{H}$ can learn anything about the roles of any other party in the network. We can conclude that for Eve in $\mathbf{H}$, Def. (33) holds for any of the subsets of $\mathcal{N}$ (cf. $\bigstar_2$ in **Tab.** E.1).

When Eve is a party in $\mathbf{P} \setminus \mathcal{A}$, the roles of the parties in either $\mathbf{P}$ or $\mathbf{C}$ are hidden because the relevant correlations of the state are unchanged by running `AME` – they essentially share a GHZ state, possibly including some additional phase, and therefore there are no revealing correlations available to anyone

but $\mathcal{A}$, meaning that here Def. 33 also holds. The exact same argument holds for Eve in $\mathbf{C}$ with respect to the anonymity of $\mathbf{P}$ (cf. $\bigstar_3$ in **Tab.** E.1).

E.2 | Anonymity during VERIFICATION

At the start of the **Verification** round, the state is the post-measurement state from (E.7), up to the correction by $\mathcal{A}$. We allow for a faulty correction, therefore keeping the phase arbitrary in the following analysis, writing $(-1)^{\Delta} = \pm 1$ for the phase. We again calculate the probability that, based on some basis choice $\{b_i\}$ and given the AME measurement outcome $x^{\alpha}_{\mathbf{H}}$, the measurement outcome $\mu^{\alpha} = \{\mu_j \mid j \in \mathbf{P} \setminus \mathcal{A}\}$ takes some particular value $o^{\alpha} = \{o^{\alpha}_i\} \in \{0,1\}^{|\mathbf{P}\setminus\mathcal{A}|}$, show that the outcome is uniformly random and that there are no correlations between the outcome and the quantum states of all possible Eves. That is, we want to show that

$$\Pr\left(\mu^{\alpha} = o^{\alpha} \mid \mathcal{I}^{+}_{\mathrm{Eve}}, \mathcal{I}_{\mathrm{Eve}}\right) = \Pr\left(\mu^{\alpha} = o^{\alpha}\right) = \frac{1}{2^{|\mathbf{P}\setminus\mathcal{A}|}}, \tag{E.8}$$

where Eve may be anyone in $\mathbf{P} \setminus \mathcal{A}$, $\mathbf{H}$ or $\mathbf{C}$. Again, we also show that the post-measurement states do not possess any other correlations regarding the roles of the parties which are exploitable by anyone in $\mathbf{P} \setminus \mathcal{A}$, $\mathbf{H}$ or $\mathbf{C}$.

Each measurement outcome is associated with a certain measurement projector $O^{\alpha}_{\mathbf{P}\setminus\mathcal{A}}$, which is itself dependent on the basis choice $\{b_i\}$. Explicitly, we define

$$\begin{aligned} O^{\alpha}_{\mathbf{P}\setminus\mathcal{A}}(\{b_i\}) := &\left(\bigotimes_{\{i\in\mathbf{P}\setminus\mathcal{A}\mid b_i=0\}} H_i \left|o^{\alpha}_i\right\rangle\!\left\langle o^{\alpha}_i\right| H_i\right) \\ &\otimes \left(\bigotimes_{\{i\in\mathbf{P}\setminus\mathcal{A}\mid b_i=1\}} \sqrt{Z_i} H_i \left|o^{\alpha}_i\right\rangle\!\left\langle o^{\alpha}_i\right| H_i \sqrt{Z_i}^{\dagger}\right). \end{aligned} \tag{E.9}$$

Hence, for any outcome $x^{\alpha}_{\mathbf{H}}$ during AME, the probability of the measurement outcome μ^{α} being equal to o^{α} becomes (remember that Δ may depend on

$x^{\alpha}_{\mathbf{H}}$)

$$
\begin{aligned}
\Pr\left(\mu^{\alpha}=m^{\alpha}\right) =& \operatorname{tr}\left[O^{\alpha}\left|\mathcal{N}_{\mathtt{postAME}}\right\rangle\!\left\langle\mathcal{N}_{\mathtt{postAME}}\right|\right] \\
=& \frac{1}{2}\operatorname{tr}\left[\left|0\right\rangle\!\left\langle 0\right|_{\mathcal{A}}\right]\operatorname{tr}\left[O^{\alpha}\left|0\ldots0\right\rangle\!\left\langle 0\ldots0\right|_{\mathbf{P}\setminus\mathcal{A}}\right]\operatorname{tr}\left[\left|\mathbf{H}\right\rangle\!\left\langle\mathbf{H}\right|\right]\otimes\left|\Psi\right\rangle\!\left\langle\Psi\right|_{\mathbf{C}} \\
+(-1)^{\Delta}& \frac{1}{2}\operatorname{tr}\left[\left|0\right\rangle\!\left\langle 1\right|_{\mathcal{A}}\right]\operatorname{tr}\left[O^{\alpha}\left|0\ldots0\right\rangle\!\left\langle 1\ldots1\right|_{\mathbf{P}\setminus\mathcal{A}}\right]\operatorname{tr}\left[\left|\mathbf{H}\right\rangle\!\left\langle\mathbf{H}\right|\right]\otimes\left|\Psi\right\rangle\!\left\langle\Phi\right|_{\mathbf{C}} \\
+(-1)^{\Delta}& \frac{1}{2}\operatorname{tr}\left[\left|1\right\rangle\!\left\langle 0\right|_{\mathcal{A}}\right]\operatorname{tr}\left[O^{\alpha}\left|1\ldots1\right\rangle\!\left\langle 0\ldots0\right|_{\mathbf{P}\setminus\mathcal{A}}\right]\operatorname{tr}\left[\left|\mathbf{H}\right\rangle\!\left\langle\mathbf{H}\right|\right]\otimes\left|\Phi\right\rangle\!\left\langle\Psi\right|_{\mathbf{C}} \\
+& \frac{1}{2}\operatorname{tr}\left[\left|1\right\rangle\!\left\langle 1\right|_{\mathcal{A}}\right]\operatorname{tr}\left[O^{\alpha}\left|1\ldots1\right\rangle\!\left\langle 1\ldots1\right|_{\mathbf{P}\setminus\mathcal{A}}\right]\operatorname{tr}\left[\left|\mathbf{H}\right\rangle\!\left\langle\mathbf{H}\right|\right]\otimes\left|\Phi\right\rangle\!\left\langle\Phi\right|_{\mathbf{C}} \\
=& \frac{1}{2}\operatorname{tr}\left[O^{\alpha}\left|0\ldots0\right\rangle\!\left\langle 0\ldots0\right|_{\mathbf{P}\setminus\mathcal{A}}\right] \\
+& \frac{1}{2}\operatorname{tr}\left[O^{\alpha}\left|1\ldots1\right\rangle\!\left\langle 1\ldots1\right|_{\mathbf{P}\setminus\mathcal{A}}\right].
\end{aligned}
\tag{E.10}
$$

Substituting O^{α} we obtain

$$
\begin{aligned}
\Pr\left(\mu^{\alpha}=m^{\alpha}\right) =& \frac{1}{2}\prod_{\{i\in\mathbf{P}\setminus\mathcal{A}\mid b_i=0\}}\langle o_i^{\alpha}|H_i|0\rangle\ \langle 0|H_i|o_i^{\alpha}\rangle \\
& \prod_{\{i\in\mathbf{P}\setminus\mathcal{A}\mid b_i=1\}}\langle o_i^{\alpha}|H_i\sqrt{Z_i}^{\dagger}|0\rangle\ \langle 0|\sqrt{Z_i}H_i|o_i^{\alpha}\rangle \\
+\frac{1}{2}& \prod_{\{i\in\mathbf{P}\setminus\mathcal{A}\mid b_i=0\}}\langle o_i^{\alpha}|H_i|1\rangle\ \langle 1|H_i|o_i^{\alpha}\rangle \\
& \prod_{\{i\in\mathbf{P}\setminus\mathcal{A}\mid b_i=1\}}\langle o_i^{\alpha}|H_i\sqrt{Z_i}^{\dagger}|1\rangle\ \langle 1|\sqrt{Z_i}H_i|o_i^{\alpha}\rangle \\
=& \frac{1}{2}\prod_{\{i\in\mathbf{P}\setminus\mathcal{A}\mid b_i=0\}}|\langle o_i^{\alpha}|+\rangle|^2\prod_{\{i\in\mathbf{P}\setminus\mathcal{A}\mid b_i=1\}}|\langle o_i^{\alpha}|+\rangle|^2 \\
+\frac{1}{2}& \prod_{\{i\in\mathbf{P}\setminus\mathcal{A}\mid b_i=0\}}|\langle o_i^{\alpha}|-\rangle|^2\prod_{\{i\in\mathbf{P}\setminus\mathcal{A}\mid b_i=1\}}|\langle o_i^{\alpha}|-\rangle|^2 \\
=& \frac{1}{2^{|\mathbf{P}\setminus\mathcal{A}|}},
\end{aligned}
\tag{E.11}
$$

which satisfies the second equation in Eq. (E.8). The global post-measurement

state ρ_{postVER} becomes

$$\begin{aligned}
\rho_{\text{postVER}} =& O^\alpha \left|\mathcal{N}_{\text{postAME}}\right\rangle\!\left\langle\mathcal{N}_{\text{postAME}}\right| O^\alpha \\
=& \frac{1}{2} \left|0\right\rangle\!\left\langle 0\right|_{\mathcal{A}} \otimes \left(O^\alpha \left|0\dots0\right\rangle\!\left\langle 0\dots0\right|_{\mathbf{P}\setminus\mathcal{A}} O^\alpha\right) \otimes \left|\mathbf{H}\right\rangle\!\left\langle\mathbf{H}\right| \otimes \left|\Psi\right\rangle\!\left\langle\Psi\right|_{\mathbf{C}} \\
+(-1)^{\Delta} & \frac{1}{2} \left|0\right\rangle\!\left\langle 1\right|_{\mathcal{A}} \otimes \left(O^\alpha \left|0\dots0\right\rangle\!\left\langle 1\dots1\right|_{\mathbf{P}\setminus\mathcal{A}} O^\alpha\right) \otimes \left|\mathbf{H}\right\rangle\!\left\langle\mathbf{H}\right| \otimes \left|\Psi\right\rangle\!\left\langle\Phi\right|_{\mathbf{C}} \\
+(-1)^{\Delta} & \frac{1}{2} \left|1\right\rangle\!\left\langle 0\right|_{\mathcal{A}} \otimes \left(O^\alpha \left|1\dots1\right\rangle\!\left\langle 0\dots0\right|_{\mathbf{P}\setminus\mathcal{A}} O^\alpha\right) \otimes \left|\mathbf{H}\right\rangle\!\left\langle\mathbf{H}\right| \otimes \left|\Phi\right\rangle\!\left\langle\Psi\right|_{\mathbf{C}} \\
+& \frac{1}{2} \left|1\right\rangle\!\left\langle 1\right|_{\mathcal{A}} \otimes \left(O^\alpha \left|1\dots1\right\rangle\!\left\langle 1\dots1\right|_{\mathbf{P}\setminus\mathcal{A}} O^\alpha\right) \otimes \left|\mathbf{H}\right\rangle\!\left\langle\mathbf{H}\right| \otimes \left|\Phi\right\rangle\!\left\langle\Phi\right|_{\mathbf{C}} \\
=& \frac{1}{2} \left|0\right\rangle\!\left\langle 0\right|_{\mathcal{A}} \otimes \left|\mathbf{P}\setminus\mathcal{A}\right\rangle\!\left\langle\mathbf{P}\setminus\mathcal{A}\right| \otimes \left|\mathbf{H}\right\rangle\!\left\langle\mathbf{H}\right| \otimes \left|\Psi\right\rangle\!\left\langle\Psi\right|_{\mathbf{C}} \\
+\gamma^\dagger & \frac{1}{2} \left|0\right\rangle\!\left\langle 1\right|_{\mathcal{A}} \otimes \left|\mathbf{P}\setminus\mathcal{A}\right\rangle\!\left\langle\mathbf{P}\setminus\mathcal{A}\right| \otimes \left|\mathbf{H}\right\rangle\!\left\langle\mathbf{H}\right| \otimes \left|\Psi\right\rangle\!\left\langle\Phi\right|_{\mathbf{C}} \\
+\gamma & \frac{1}{2} \left|1\right\rangle\!\left\langle 0\right|_{\mathcal{A}} \otimes \left|\mathbf{P}\setminus\mathcal{A}\right\rangle\!\left\langle\mathbf{P}\setminus\mathcal{A}\right| \otimes \left|\mathbf{H}\right\rangle\!\left\langle\mathbf{H}\right| \otimes \left|\Phi\right\rangle\!\left\langle\Psi\right|_{\mathbf{C}} \\
+& \frac{1}{2} \left|1\right\rangle\!\left\langle 1\right|_{\mathcal{A}} \otimes \left|\mathbf{P}\setminus\mathcal{A}\right\rangle\!\left\langle\mathbf{P}\setminus\mathcal{A}\right| \otimes \left|\mathbf{H}\right\rangle\!\left\langle\mathbf{H}\right| \otimes \left|\Phi\right\rangle\!\left\langle\Phi\right|_{\mathbf{C}} \Big] \\
=& \left|\mathcal{N}_{\text{postVER}}\right\rangle\!\left\langle\mathcal{N}_{\text{postVER}}\right| ,
\end{aligned} \tag{E.12}$$

where $\gamma = (-1)^{\Delta} \times (-i)^{|\{b_i\}|}$ and $\left|\mathcal{N}_{\text{postVER}}\right\rangle$ is the pure state

$$\left|\mathcal{N}_{\text{postVER}}\right\rangle := \left(\left|0\right\rangle_{\mathcal{A}} \otimes \left|\Psi\right\rangle_{\mathbf{C}} + \gamma \left|1\right\rangle_{\mathcal{A}} \otimes \left|\Phi\right\rangle_{\mathbf{C}}\right) \otimes \left|\mathbf{P}\setminus\mathcal{A}\right\rangle \otimes \left|\mathbf{H}\right\rangle \tag{E.13}$$

and $\left|\mathbf{P}\setminus\mathcal{A}\right\rangle$ is the state associated with the measurement outcome o^α

$$\left|\mathbf{P}\setminus\mathcal{A}\right\rangle := \left(\bigotimes_{i\in\{\mathbf{P}\setminus\mathcal{A}|b_i=0\}} H_i \left|o_i^\alpha\right\rangle_i\right) \otimes \left(\bigotimes_{i\in\{\mathbf{P}\setminus\mathcal{A}|b_i=1\}} \sqrt{Z_i} H_i \left|o_i^\alpha\right\rangle_i\right) . \tag{E.14}$$

From the perspective of $\mathbf{H}$, all communication is indistinguishable (cf. the VERIFICATION column in **Tab.** E.2); $\mathbf{H}$ is dis-entangled from everyone else and the state on $\mathbf{H}$ is itself separable. We can conclude that – with anyone in $\mathbf{H}$ as Eve – the anonymity of everyone in the network is preserved (cf. $\checkmark_1$ in **Tab.** E.1).

Moreover, $\mathbf{P}\setminus\mathcal{A}$ is dis-entangled from all other parties in the network and their post-measurement state is separable as well. Again, all communication from their perspective is uniformly random (cf. the VERIFICATION column in Tab. E.2), so we can conclude that – with anyone in $\mathbf{P}\setminus\mathcal{A}$ as Eve – the anonymity of everyone in the network is maintained (cf. $\checkmark_2$ in **Tab.** E.1).

The only relevant information is $|\{b_i\}|$, which is encoded into the phase of the state on $\mathcal{A}\cup\mathbf{C}$; any phase estimation algorithm to retrieve this information would require access to the entire state, including the state of $\mathcal{A}$, which is

inaccessible to **C**. Again, from the perspective of **C** all communication is indistinguishable (cf. the VERIFICATION column in **Tab.** E.2) and we can conclude that – with **C** as Eve – here too the anonymity of all parties in the network is preserved (cf. $\checkmark_3$ in **Tab.** E.1).

Note that the **Verification** round can only pass if $|\Psi\rangle_{\mathbf{C}} = |\Phi\rangle_{\mathbf{C}}$, that is when **C** is not entangled to $\mathcal{A}$ and $\mathbf{P} \setminus \mathcal{A}$. However, this is not a necessary condition for anonymity, since the identity of Alice is preserved even if the **Verification** round fails. There is no information encoded into the state regarding the distribution of **P** and **H**, nor into the measurement outcome o^{α}. The only valuable information in the state is the parity of the number of Y-measurements, encoded in the phase of the qubit of $\mathcal{A}$, which is dis-entangled from all other parties and therefore only accessible to $\mathcal{A}$.

E.3 | Anonymity during the KeyGen rounds

As the **Verification** rounds ensure that the $|\mathrm{GHZ}_{m+1}\rangle$ state on **P** is dis-entangled from the non-participating parties in $\bar{\mathbf{P}}$ and after running AME no party in **H** is entangled to any other party, all subsets listed in **Tab.** E.1 are dis-entangled from each other. Hence, we can write the full-network state at the start of the **Keygen** round as

$$|\mathcal{N}_{\mathbf{Keygen}}\rangle \hat{=} |\mathrm{GHZ}\rangle_{\mathbf{P}} \otimes |\mathbf{H}\rangle \otimes |\Psi\rangle_{\mathbf{C}} \,. \tag{E.15}$$

Since there is no communication during the **Keygen** rounds, there is no leakage from **P**, **H**, **C** outside the subset itself (cf. $\multimap_1$ in **Tab.** E.1). As $|\mathbf{H}\rangle$ is a separable state, the case **H** is trivial (cf. $\multimap_2$ in **Tab.** E.1). Finally, due to its symmetries, the $|\mathrm{GHZ}_{m+1}\rangle$ state cannot reveal who the parties sharing the state are. This ensures that there is no privacy leakage for **P** either (cf. $\multimap_3$ in **Tab.** E.1).

F

Corrections during LinACKA

This appendix details the corrections that the participants Alice, Bob and Charlie have to perform on their qubits during `LinACKA` as presented in chapter 9. It is directly sourced from Pub. [D], where it is included as appendix *A*. Following the original presentation there, Alice, Bob and Charlie are referred to as a, b and c instead of $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$.

The corrections are divided into three separate parts: the *configuration corrections*, that depend on the locations of the participants in the linear network, and the two corrections that depend on the measurement outcomes of the non-participants, the *X-correction* and the *Z-correction*.

F.1 | Detailing the necessary corrections

Alice and Charlie need to perform a correction to obtain the $|\mathrm{GHZ}_3\rangle$ state with Bob, whereas Bob never has to perform a non-trivial rotation. The corrections for Alice and Charlie are structurally similar; we first introduce those for Alice. In order to achieve this, we define the following quantities.

- $\delta_{ab} := b - a - 1$, the number of non-participants between Alice and Bob.
- $p_{ab} := \delta_{ab} \mod 4$, the mod-four value of δ_{ab}
- $g_{ab} := \frac{\delta_{ab} - p_{ab}}{4}$, the integer number of groups of four that fit between Alice and Bob.

For Charlie, δ_{cb}, g_{cb} and p_{cb} are defined in a similar fashion. We refer to Fig. F.1 for two potential configurations of the network that exemplifies these definitions.
Alice now performs the following correction steps:

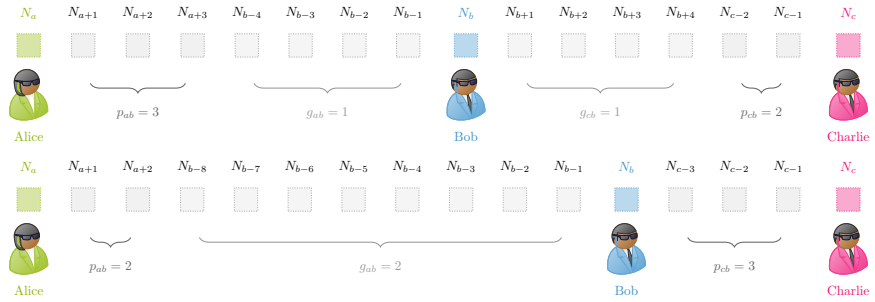


Figure F.1: Two exemplary configurations. **Top:** $\delta_{ab} = 7$ (with $p_{ab} = 3$ and $g_{ab} = 1$) and $\delta_{cb} = 6$ (with $p_{cb} = 2$ and $g_{cb} = 1$). **Bottom:** $\delta_{ab} = 10$ (with $p_{ab} = 2$ and $g_{ab} = 2$) and $\delta_{cb} = 3$ (with $p_{cb} = 3$ and $g_{cb} = 0$).

1. Apply a *configuration* correction C_{ab} depending on p_{ab} and g_{ab}, as shown in Tab. F.1, picking the left (brown, $\beta_a = 1$) or right (green, $\beta_a = 0$) table.
2. Divide all the measurement outcomes $\{m_i\}_{a+1}^{b-1}$ into a set $\{m_i\}_{a+1}^{a+1+p_{ab}}$ and a set $\{m_i\}_{a+2+p_{ab}}^{b-1}$ – where it is to be understood that if $p_{ab} = 0$, the first set is empty.
3. From the outcomes in the first set, they calculate the bits k_x and k_z using Tab. F.1.
4. From the outcomes in the second set, out of every pair of four they select the measurement outcomes as described in Tab. F.2 and add them all together to calculate l_x and l_z, respectively (e.g. if $\beta_a = 1$, Alice selects every odd element of the second set to calculate l_x, and every second, third and fourth out of four to calculate l_z).
5. They apply an X operation on their qubit if and only if $k_x \oplus l_x = 1$.
6. They apply a Z operation on their qubit if and only if $k_z \oplus l_z = 1$.

Note that all three corrections (i.e. the configuration correction, the X correction and the Z correction) can be contracted into a single Clifford operation. However, since the measurement-outcome dependent corrections are only Pauli operators, they will at most flip the measurement outcomes for Alice in the subsequent steps of the protocol – and need not be physically implemented. This also means that the participants can perform their `KeyGen` or `Verification` measurements *before* the measurement outcomes of the non-participants are announced. By having all nodes $\{a+1, \ldots, b-1\}$ perform their measurements and Alice subsequently perform the aforementioned corrections, the linear cluster state is contracted towards a $|L_{a,b,b+1,\ldots,c-1,c}\rangle$ linear cluster state. Hence, Charlie can perform the same steps (while using the

δ_{ab} mod 4	C_{ab}	k_x	k_z
0	$Z^{g_{ab}}$	✗	✗
1	$Z^{g_{ab}}H$	✗	m_{a+1}
2	$Z^{g_{ab}}P_z$	m_{a+1}	$m_{a+1} \oplus m_{a+2}$
3	$Z^{g_{ab}}HP_z$	$m_{a+1} \oplus m_{a+2}$	$m_{a+1} \oplus m_{a+3}$

δ_{ab} mod 4	C_{ab}	k_x	k_z
0	$X^{g_{ab}}$	✗	✗
1	$X^{g_{ab}}HP_x$	✗	m_{a+1}
2	$X^{g_{ab}}P_x$	m_{a+1}	m_{a+2}
3	$X^{g_{ab}}HX$	m_{a+2}	$m_{a+1} \oplus m_{a+2} \oplus m_{a+3}$

Table F.1: Local corrections that Alice needs to perform to obtain the GHZ state with Bob and Charlie after the non-participants measured their qubits. The left table shows the corrections if the non-participant $a+1$ after Alice measured in the X-basis ($\beta_a = 1$), the right table the corrections if it was in the Y-basis ($\beta_a = 0$). **The C_{ab} column** contains the *configuration* correction which only depends on the number of non-participants δ_{ab} between Alice and Bob – note that $g_{ab} := \lfloor \delta_{ab}/4 \rfloor$. **The k_x column** contains the measurement outcomes that add to k_x, which induce together with l_x a correction $X^{k_x \oplus l_x}$; similarly **the k_z column** contains the measurement outcomes that create k_z.

	$\beta_a = 1$	$\beta_a = 0$
l_x	1st, 3rd	1st, 2nd, 3rd
l_z	2nd, 3rd, 4th	2nd, 4th

Table F.2: Selection of measurement outcomes out of every pair of four from the second set to calculate l_x and l_z, respectively. For example, when $\delta_{ab} = 7$ and $\beta_a = 1$, $l_x = m_{a+4} \oplus m_{a+6}$ and $l_z = m_{a+5} \oplus m_{a+6} \oplus m_{a+7}$.

measurement outcomes $\{m_{c-1}, m_{c-2}, \ldots, m_{b+1}\}$, $\delta_{bc} := c - b - 1$ and its redefined derivatives) to contract the state towards a three-partite linear cluster state $|L_{a,b,c}\rangle$. Two final H gates for Alice and Charlie result in the desired $|\mathrm{GHZ}_3\rangle$ state between Alice, Bob and Charlie.

F.2 Calculating the corrections

Using the stabilizer formalism, it is straightforward to show that, starting from a linear cluster state $|L_{a,a+1,\ldots,c-1,c}\rangle$, a measurement on node $a+1$ in the X- or Z-basis results in $|L_{a,a+2,\ldots,c-1,c}\rangle$ up to a local correction C_{ab}^{a+1} for Alice, where this correction depends on both the measurement basis β_{a+1} and outcome m_{a+1} as

$$C_{ab}^{a+1}(m_{a+1}, \beta_{a+1}) = P_z^{(2m_{a+1}+\beta_{a+1})} H = H P_x^{(2m_{a+1}+\beta_{a+1})}, \tag{F.1}$$

where $P_z := R_z\left(\frac{\pi}{2}\right)$ is a half-rotation around the Z-axis and P_x is defined similarly. Note that either identity (i.e. the Z- or X-based rotation) can be used.

A series of multiple measurements then introduces a concatenation of these corrections, where the corrections are performed in order from $a+1$ to $b-1$. They do not necessarily commute, but by using the X- and Z-based correction interchangeably (and thus cancelling out the H operations), and using the identity $Z^{m_i}P_x = P_x X^{m_i} Z^{m_i}$ (and likewise for P_z) one can group all the corrections that are not measurement outcome based together as the first corrections; this allows to partition the complete correction into a 'configuration' correction and an outcome-based correction.

Specifically, for the alternating pattern of X-basis and Y-basis measurements, each group of four consecutive measurements together introduces only Pauli corrections. For example, for any group of four consecutive nodes $\{1,2,3,4\}$ (note that these labels resemble *any* set of four consecutive nodes) these corrections are

$$X^{(m_1+m_3)} Z^{(m_2+m_3+m_4)} X, \qquad (\beta_a = 1)$$
$$X^{(m_1+m_2+m_3)} Z^{(m_2+m_4)} Z. \qquad (\beta_a = 0)$$

Up to an irrelevant global phase, all these operators commute with each other. Therefore, starting from the last measured node (i.e. $b-1$) an integer multiple of four can be 'stitched together'. Since there are $g_{ab} := \lfloor \delta_{ab}/4 \rfloor$ of such groups, the correction becomes

$$\prod_{i=0}^{g_{ab}-1} X^{(m_{b-4i-4} \oplus m_{b-4i-2})} Z^{(m_{b-4i-3} \oplus m_{b-4i-2} \oplus m_{b-4i-1})} X = X^{l_x} Z^{l_z} X^{g_{ab}}, \qquad (\beta_a = 1)$$

$$\prod_{i=0}^{g_{ab}-1} X^{(m_{b-4i-4} \oplus m_{b-4i-3} \oplus m_{b-4i-2})} Z^{(m_{b-4i-3} \oplus m_{b-4i-1})} X = X^{l_x} Z^{l_z} Z^{g_{ab}}, \qquad (\beta_a = 0)$$

where l_x is defined as

$$l_x := \bigoplus_{i=0}^{g_{ab}-1} m_{b-4i-4} \oplus m_{b-4i-2}, \qquad (\beta_a = 1)$$
$$l_x := \bigoplus_{i=0}^{g_{ab}-1} m_{b-4i-4} \oplus m_{b-4i-3} \oplus m_{b-4i-2}, \qquad (\beta_a = 0)$$

and l_z is defined as

$$l_z := \bigoplus_{i=0}^{g_{ab}-1} m_{b-4i-3} \oplus m_{b-4i-2} \oplus m_{b-4i-1}, \qquad (\beta_a = 1)$$
$$l_z := \bigoplus_{i=0}^{g_{ab}-1} m_{b-4i-3} \oplus m_{b-4i-1}. \qquad (\beta_a = 0)$$

The corrections for the measurements of the nodes $a+1, \ldots, a+p_{ab}$ (i.e. the first p_{ab} measurements) are then also grouped together; by splitting them into a measurement-outcome dependent and -independent part, they can be written as $X^{k_x} Z^{k_z} C_{ab}$, where C_{ab}, k_x and k_z can be read from Tab. F.1. Note that the $X^{g_{ab}}$ or $Z^{g_{ab}}$ in Tab. F.1 is technically not part of the correction here, but that they will commute with $X^{k_x} Z^{k_z}$ and hence the total correction that Alice needs to perform becomes (where now C_{ab} is as in Tab. F.1):

$$X^{l_x} Z^{l_z} X^{k_x} Z^{k_z} C_{ab} \doteq X^{k_x \oplus l_x} Z^{k_z \oplus l_z} C_{ab}, \tag{F.2}$$

where $\doteq$ here indicates 'up to an (irrelevant) global phase'. Since these corrections only consider nodes between Alice and Bob, and since there are actions that Bob needs to perform, the corrections for Charlie work in a similar fashion and can be seen separately from these.

G

Security proof of LinACKA

This appendix contains a statement of LinACKA and a security proof of the generated key. Note that the preparation of the network state $|\mathcal{N}\rangle$ (i.e. Protocol III) has been omitted, as it does not affect security. This appendix is directly sourced from Pub. **[D]** ([46]) where it is included as appendix B.

Although the security is proven under Def. 29 and Def. 30, the definitions are restated in this appendix for easy reference. It should be noted that the security of the protocol is proven under even less restrictive assumptions than introduced in chapter 9, because it allows for collective attacks by multiple non-participants together.

G.1 | Protocol statement

Input:

- L network states $|\mathcal{N}\rangle$ connecting $\{i\}_{i=1}^{n}$, including $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$.
- Desired secrecy parameter $\varepsilon_{\mathrm{s}} > 0$, which determines a correlation threshold Q_{tol}, and correctness parameter $\varepsilon_{\mathrm{c}} > 0$.
- A random string s_{b} of length $L \cdot h_2(p)$ secretly pre-shared between the participants to randomly choose m out of the L cluster states to be measured in the X-basis for parameter estimation where $p = m/L$, leaving $k := L - m$ measurements in the Z-basis for key generation.
- An estimate of the expected bit error rate Q_z in the Z-basis between Alice and Bob and Alice and Charlie. The worst of these will be used to select an error-correcting code that requires an error syndrome of length $\ell_{\mathrm{EC}} := k \cdot h_2(Q_z)$ to be announced.

- A pre-shared secret random string s_{EC} of length ℓ_{EC} to be used to one-time pad the error reconciliation announcements, another pre-shared string s_{hEC} of length $\ell_{\mathrm{hEC}} := \log_2(2/\varepsilon_{\mathrm{c}})$ to one-time pad the error correction verification announcements, and three bits of pre-shared key to communicate aborting by the participants.
- Two pre-shared random strings, s_{h} and s_{hEC}, of lengths $k+\ell_{\mathrm{PA}}-1$ and $k+\ell_{\mathrm{hEC}}-1$ respectively to be used as the seeds for hashing, where ℓ_{PA} is the output of the privacy amplification hashing as defined below. The string s_{h} is used for privacy-amplification of the private key, while s_{hEC} is used to verify the error correction step has succeeded. Note that unlike the previous seeds, these can be used indefinitely and need not be replenished after each run of the protocol.

Output: A key of length ℓ shared anonymously between Alice, Bob and Charlie that is ε_{s}-secret and ε_{c}-correct.

1. For $i = 1, \ldots, n$:
   (a) Node i receives bit β_{i-1} and computes $\beta_i = 1 - \beta_{i-1}$, except for 1 who draws a random bit β_0 instead.
      i. If $i \in \bar{\mathbf{P}}$, they measure the operator X^i or Y^i if $\beta_i = 0$ or $\beta_i = 1$, respectively. They broadcast the measurement outcome m_i.
      ii. If $i \in \mathbf{P}$, they announce a uniform randomly drawn bit m_i.
   (b) Node i sends bit β_i to neighbour $i+1$, except for node n.
2. The participants perform corrections on their qubits to obtain the desired $|\mathrm{GHZ}_n\rangle$ state.
   (a) Alice and Charlie apply their *configuration corrections* C_a and C_c, respectively (**Tab.** F.1).
   (b) Alice ($i = a$) and Charlie ($i = c$) both calculate their parameters l_x^i, k_x^i and l_z^i, k_z^i from the measurement outcomes of the non-participants (**Tabs.** F.1 and F.2) and each apply $X^{l_x \oplus k_x}$ and $Z^{l_z \oplus k_z}$ to their qubit.
   (c) Alice and Charlie each apply a Hadamard operation H to their qubit to obtain the final desired $|\mathrm{GHZ}_n\rangle$ state.
3. Using the pre-shared string s_b, the participants coordinate their measurements of all L $|\mathrm{GHZ}_n\rangle$ states into m `Verification` rounds (i.e. X-basis) and k `KeyGen` rounds (i.e. Z-basis). Everyone announces after each measurement a random bit m_i, except for Bob and Charlie, who announce their measurement outcomes for the `Verification` rounds.

4. Alice, who can locate Bob's and Charlie's measurement outcomes from the `Verification` rounds, estimates the X-basis error rate $Q_X^m = \frac{1}{2}(1 - \langle X^a X^b X^c \rangle)$. If this is above the *tolerance* Q_{tol}, she aborts by setting her *abort bit* to 1.

5. Alice computes the necessary information for error correction – the *error syndrome* of length ℓ_{EC} – and then one-time pad encrypts this information with the string s_{EC}. All other players announce uniform random strings of length ℓ_{EC}.

6. Bob and Charlie use their copies of s_{EC} to obtain l_{EC} and correct their k Z measurement strings, i.e. their raw key. Alice, Bob and Charlie then hash their string using the seed s_{hEC}. Alice encrypts her output using her copy of s_{hEC}. Using their copy, Bob and Charlie each decrypt Alice's hash outcome and compare it to their own; if they do not align, they abort by setting their *abort* bit to 1.

7. Alice, Bob and Charlie, using another three bits of the pre-shared key, encrypt their *abort bit* – which is equal to 1 if and only if they want to abort – and announce it, while all other parties announce uniformly random bits instead. If any participants announced a 1, everyone aborts (meaning they will not use the generated key).

8. Finally, the participants hash their measurement results with the seed s_{hPA} to produce the final key s of length

$$\begin{aligned} l_{\text{PA}} :=& k \left[1 - h_2 \left(Q_{\text{tol}} + \mu \left(\frac{\varepsilon_{\text{s}} - \varepsilon}{2} \right) \right) \right] + 2 + 2 \log_2(\varepsilon) \\ =& \ell + \ell_{\text{EC}} + \ell_{\text{hEC}} + L \cdot h_2(p) + 3. \end{aligned}$$

   However, to fairly evaluate performance the parties should replenish their stock of secret-shared key so as to be able to perform subsequent CKA protocols. Subtracting off the non-reusable shared randomness results in a string of length ℓ that is available for applications.

G.2 | Security proof

We now prove the security of our protocol in the scope of an even more general adversary model than the one introduced in the main text, so that we can resort to a powerful machinery that has been developed in the literature [175, 214] and we can build on the strategy of proof laid out in Ref. [215]; the security of our protocol within our adversary model then follows readily. However, there are some variations to the tools necessary to preserve the anonymity of the participants which is key to the present work. We briefly explain some critical quantities and definitions. Let $\rho_{S_A S_B S_C E'}$ be the

joint classical-quantum state between the final keys of the participants and an eavesdropper conditioned on passing all checks. Note that the eavesdroppers system, $E' = ER$, is made up of a quantum system, E, that completely purifies the pre-measurement state ρ_{ABC} (and is, therefore, assumed to include system of the non-participating player) and a classical register R that contains all of the information announced during the protocol. A protocol is called ε_{rob}-robust if it passes the correlation and the error correction checks with probability $1 - \varepsilon_{\text{rob}}$. Defining a uniformly distributed state as

$$\rho_{\mathbf{U}} \equiv \sum_{s \in \mathcal{S}} \frac{1}{|\mathcal{S}|} |s\rangle\langle s| \tag{G.1}$$

with $\mathcal{S}$ the set of possible secret keys we have the following definition [215].

Definition 34 (Approximate robustness and secrecy)**.** *A CKA protocol that is ε_{rob}-robust is ε_c-correct if*

$$(1 - \varepsilon_{\text{rob}}) \Pr\left[S_A \neq S_B \vee S_A \neq S_C\right] \leqslant \varepsilon_c \tag{G.2}$$

and ε_s-secret if

$$(1 - \varepsilon_{\text{rob}}) \frac{1}{2} \left\| \rho_{S_A E'} - \rho_{\text{U}} \otimes \rho_{E'} \right\| \leqslant \varepsilon_s \tag{G.3}$$

is called $(\varepsilon_s + \varepsilon_c)$-secure if it is ε_c-correct and ε_c secret.

Turning first to multi-partite error correction we have the following statement.

Theorem 4 (Theorem 2 in Ref. [215])**.** *Given a probability distribution $P_{X_A, B_1, B_2, \ldots, B_N}$, between Alice and n other players there exists a one-way error-correction protocol for all n players that is: ε_c-correct, and $2(n-1)\varepsilon'$ -robust on $P_{X_A, B_1, B_2, \ldots, B_n}$, and has leakage*

$$\ell_{\text{EC}} \leqslant \max_i H_0^{\varepsilon'}(X_A|B_i) + \log_2 \frac{2(n-1)}{\varepsilon_c}. \tag{G.4}$$

In terms of secrecy the critical results are leftover hashing against quantum side-information, an entropic uncertainty relation for smoothed min- and max-entropies, applied to our protocol, states the following.

Lemma 1 (Leftover hashing against quantum side information in Refs. [183, 214])**.** *Let $\varepsilon' \geq 0$ and $\rho_{Z_A E}$ be a classical-quantum state where Z_A is defined over a discrete-valued and finite alphabet, E is a quantum system and R is a register containing the classical information learnt by Eve during information reconciliation. If Alice applies a hash function, drawn at random from a*

family of two-universal hash functions that maps $\rho_{Z_A E}$ to $\rho_{S_A E}$ and generates a string of length ℓ, then

$$\frac{1}{2}\left\|\rho_{S_A E R} - \rho_{\mathrm{U}} \otimes \rho_{E R}\right\| \leqslant 2^{-\frac{1}{2}\left(H_{\min}^{\varepsilon'}(Z_A|ER)-\ell+2\right)} + 2\varepsilon', \tag{G.5}$$

where $H_{\min}^{\varepsilon'}(Z_A|ER)$ is the conditional smooth min-entropy of the raw measurement data given Eve's quantum system and the leakage of the information reconciliation.

This leads to the following corollary.

Corollary 3 (Secret string extraction)**.** *For an $\varepsilon_{\mathrm{rob}}$-robust protocol an ε_s-secret string of length*

$$\ell = H_{\min}^{\varepsilon'}(Z_A|ER) + 2 - 2\log_2\frac{1}{\varepsilon} \tag{G.6}$$

can be extracted for any $\varepsilon_s, \varepsilon, \varepsilon' \geq 0$ such that

$$\varepsilon_s \geq \varepsilon + 2\varepsilon' \tag{G.7}$$

where $H_{\min}^{\varepsilon'}(Z_A|ER)$ is the conditional smooth min-entropy of the raw measurement data given Eve's quantum system and the information reconciliation leakage conditioned on the protocol not aborting.

Proof: Note that if we choose

$$\ell = H_{\min}^{\varepsilon'}(Z_A|ER) + 2 - 2\log_2\frac{(1-\varepsilon_{\mathrm{rob}})}{\varepsilon}, \tag{G.8}$$

then the right hand side of (G.5) is equal to $\varepsilon/(1-\varepsilon_{\mathrm{rob}})+2\varepsilon'$. Comparing with (G.3) in Def. 34 we see we want this expression to satisfy $\varepsilon/(1-\varepsilon_{\mathrm{rob}})+2\varepsilon' \leq \varepsilon_s/(1-\varepsilon_{\mathrm{rob}})$ so our security condition is satisfied for any $\varepsilon_s \geq \varepsilon + 2(1-\varepsilon_{\mathrm{rob}})\varepsilon'$ which is true for any $\varepsilon_s \geq \varepsilon + 2\varepsilon'$ where we used that $(1-\varepsilon_{\mathrm{rob}}) \leq 1$. Noting further that $\log_2(1-\varepsilon_{\mathrm{rob}}) \leq 0$ yields (G.6). This means that, provided the constraint in (G.7) is satisfied, the positive constant ε can be optimized over. Typically this makes little difference to the final performance and and they are commonly chosen as $\varepsilon = \varepsilon_s/2$.

Now we see that the problem has condensed to determining Eve's conditional smooth min-entropy for Z_A^k (in the following we will suppress the k superscript), the variable describing the outcome of Alice's Z measurements on the k key-generating qubits. To begin with, consider the situation before any information reconciliation is exchanged (there is no register R) so we simply have $H_{\min}^{\varepsilon'}(Z_A|E)$. Since Eve's state is taken to include that of all the non-participating players we can assume without loss of generality that there is an overall pure tripartite state between Alice, the remaining participants (which we denote B_i), and Eve. The required bound for this situation

has been derived by applying an entropic uncertainty relation [183] for the smoothed min- and max-entropies specialized to the case of observables made up of the k-fold tensor product of either Z-basis and X-basis measurements, (i.e. the observables $Z_A = Z^1 \otimes Z^2 \otimes \cdots \otimes Z^k$ and $X_A = X^1 \otimes X^2 \otimes \cdots \otimes X^k$) [214]

$$\begin{aligned} H^{\varepsilon}_{\min}(Z_A|E) + H^{\varepsilon}_{\max}(X_A|B_i) \geq k, \\ \Rightarrow H^{\varepsilon}_{\min}(Z_A|E) \geq k - H^{\varepsilon}_{\max}(X_A|B_i), \end{aligned}$$

where we have used the data processing inequality in the form $H^{\varepsilon}_{\max}(X_A|B_i) \geq H^{\varepsilon}_{\max}(X_A|B_i)$ in the second line. Naively, this bound cannot be evaluated since it is counterfactual, i.e. the k qubits are always measured in the Z-basis so we have no direct access to $H^{\varepsilon}_{\max}(X_A|B_i)$, which is the conditional max-entropy of the participants given their Pauli measurements if Alice had instead measured in the X-basis in these k rounds. However, since the parameter estimation and key generation rounds were selected at random then it has been shown that Serfling's bound can be applied to statistically bound the X correlation that would have been observed in the k key generation rounds based upon those that were actually observed in the parameter estimation rounds. This is expressed in the following result.

Lemma 2 (Lemma 3 in Ref. [214])**.** *Let k be the number of key generation rounds, m be the number of parameter estimation rounds, d_0 a threshold on the number of errors that can be observed during parameter estimation without the protocol aborting and $\varepsilon' > 0$.*

$$H^{\varepsilon}_{\max}(X_A|B_i) \leq k h_2(d_0 + \mu(\varepsilon'(1 - \varepsilon_{\text{rob}}))), \tag{G.9}$$

where $\mu(\varepsilon)$ is a correction for statistical errors:

$$\mu(\varepsilon) := \sqrt{\frac{m+k}{mk}\frac{m+1}{m}\ln\frac{1}{\varepsilon}}. \tag{G.10}$$

Putting all of these results together we can prove the following security statement.

Theorem 5 (Security statement)**.** *If the anonymous CKA protocol defined above proceeds without aborting an $(\max_{i\in\{B,C\}} \ell^i_{\text{EC}}, \varepsilon_c)$ error correction protocol and a two-universal hashing are successfully applied then an $(\varepsilon_s + \varepsilon_c)$-secure key of length*

$$\begin{aligned} \ell &= k\left[1 - h_2\left(Q_{\text{tol}} + \mu\left(\frac{\varepsilon_{\text{s}} - \varepsilon}{2}\right)\right)\right] \\ &\quad + 2 - 2\log_2\frac{1}{\varepsilon} - \ell_{\text{EC}} - \ell_{\text{hEC}} - L\cdot h_2(p) - 3 \\ &= L\left[(1-p)\left[1 - h_2\left(Q_{\text{tol}} + \mu\left(\frac{\varepsilon_{\text{s}} - \varepsilon}{2}\right)\right) - h_2\left(Q_z\right)\right] - h_2(p)\right] \\ &\quad + \log_2(\varepsilon^2\varepsilon_{\text{c}}) - 2 \end{aligned} \tag{G.11}$$

can be anonymously extracted.

Proof: At the conclusion of the protocol we can immediately apply Cor. 3 to the k round classical-quantum state $\rho^k_{Z_A E} = \mathrm{tr}_{B_i}(|\Psi_{AB_iE}\rangle\langle\Psi_{AB_iE}|)$ to extract an ε_s-secret key of length

$$\ell = H_{\min}^{\varepsilon'}(Z_A|ER) + 2 - 2\log_2\frac{1}{\varepsilon} \tag{G.12}$$

for positive constants satisfying

$$\varepsilon_s \geq \varepsilon + 2(1-\varepsilon_{\mathrm{rob}})\varepsilon'. \tag{G.13}$$

Now, because all of the communication involved in error reconciliation is one-time padded to ensure anonymity we have that $H_{\min}^{\varepsilon'}(Z_A|E,R) = H_{\min}^{\varepsilon'}(Z_A|E)$ by definition. This gives

$$\begin{aligned}
\ell \quad &= \quad H_{\min}^{\varepsilon'}(Z_A|E) + 2 - 2\log_2\frac{1}{\varepsilon} \\
&\overset{G.9}{\geq} \quad k - H_{\max}^{\varepsilon'}(X_A|B_i) + 2 - 2\log_2\frac{1}{\varepsilon} \\
&\overset{G.13}{\geq} \quad k - H_{\max}^{(\varepsilon_s-\varepsilon)/2/(1-\varepsilon_{\mathrm{rob}})}(X_A|B_i) + 2 - 2\log_2\frac{1}{\varepsilon} \\
&\overset{G.9}{\geq} \quad k - kh_2\left(Q_{\mathrm{tol}} + \mu\left(\frac{\varepsilon_s-\varepsilon}{2}\right)\right) + 2 - 2\log_2\frac{1}{\varepsilon},
\end{aligned} \tag{G.14}$$

where in the third line we have also used that $H_{\max}^{\varepsilon_1}(X|Y) \leqslant H_{\max}^{\varepsilon_2}(X|Y)$ for $\varepsilon_1 \geqslant \varepsilon_2$. This string is guaranteed to be ε_s-secret and, by Thm. 4, if the error correction process did not abort then the string is also ε_c-correct. However, this is not a fair representation of the performance of the protocol, since we had to use up the reservoir of pre-shared key for the basis choices and for one-time padding the error reconciliation information. Thus, to get the length of useable key we need to calculate how much remains after we have replenished the pre-shared strings necessary for the next protocol implementation. Subtracting off the seed for basis choices, $L \cdot h_2(p)$, and the length of the error correction information and verification, ℓ_{EC} and ℓ_{hEC} and the 3 bits for the abort step, gives (G.11).

H

Anonymity proof of LinACKA

This appendix is concerned with the anonymity of the protocol. It is directly sourced from Pub. **[D]** ([46]) where it is included as appendix C. Anonymity is defined using Def. 32. Most importantly for the analysis, it needs to be shown that all public communication – the announced measurement results – is independent of the choice of participants. This is done by showing that they are uniformly random and uncorrelated. Similar to chapter G, following the original presentation in Pub. **[D]**, Alice, Bob and Charlie are referred to as a, b and c instead of $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$.

H.1 | Proof of anonymity

In the proposed protocol, the output state $\rho_{\bar{\mathbf{P}}|abc}$ has several registers. The only non-trivial registers that need to be addressed are the ones containing the classical communication of all the measurement outcomes $\{o_i\} \cup \{m_i\}$. The reason is that the reduced quantum state of any dishonest party is the maximally mixed state, which is independent of the choice of participants, and therefore trivially fulfils Def. 32. Moreover, all other parties do not hold a quantum register by the end of the protocol.

In the remainder of this section we will show that there are no correlations between any of the announced measurement outcomes $\{o_i\} \cup \{m_i\}$, i.e. that the outcome distribution is indistinguishable from that of the uniformly drawn announcements of the nodes $1, a$ and c during `Protocols III` and `IV`. We can then conclude that we have complete anonymity, i.e. our protocol is ε_{an}-anonymous for $\varepsilon_{\text{an}} = 0$.

Since the state of the network always remains separable between the tripartition of the nodes to the left of (and including) Alice, the nodes to the right of (and including) Charlie, and the nodes between (and including) Alice and Charlie, it suffices to show that there are no correlations within the measurement announcements associated with these three separate groups. We show

this absence of correlations only for the left set, since the argument applies analogously to the other two sets. We first show this in the case of an honest-but-curious non-participant, followed by the case where a non-participant may actively deviate from the protocol.

H.1.1 | Honest-but curious setting

Consider the stabilizer of the network state after all CZ operations have been performed in Step 2a of Protocol `Protocol III`. It is generated by the following collection of operators:

$$\begin{aligned} \sigma_x^{\tau_1}\sigma_z^{\omega_2}, \\ \sigma_z^{\tau_1}\sigma_x^{\omega_2}\sigma_z^{\tau_2}, \\ \{\sigma_z^{\tau_i}\sigma_z^{\omega_{i+1}}\}_{i=2}^{a-2}, \\ \{\sigma_z^{\omega_i}\sigma_x^{\tau_i}\sigma_x^{\omega_{i+1}}\sigma_z^{\tau_{i+1}}\}_{i=2}^{a-2}, \\ \sigma_z^{\omega_{a-1}}\sigma_x^{\tau_{a-1}}\sigma_x^{\omega_a}, \\ \sigma_z^{\tau_{a-1}}\sigma_z^{\omega_a}. \end{aligned} \tag{H.1}$$

The measurement operator of all measurement outcomes together depends on β_1 as

$$M = \begin{cases} \sigma_y^{\omega_2}\sigma_x^{\tau_2}\sigma_x^{\omega_3}\sigma_x^{\tau_3}\sigma_y^{\omega_4}\sigma_x^{\tau_4}\sigma_x^{\omega_5}\sigma_x^{\tau_5}\sigma_y^{\omega_6}\ldots\sigma_x^{\tau_{a-1}}, & (\beta_1 = 0) \\ \sigma_x^{\omega_2}\sigma_x^{\tau_2}\sigma_y^{\omega_3}\sigma_x^{\tau_3}\sigma_x^{\omega_4}\sigma_x^{\tau_4}\sigma_y^{\omega_5}\sigma_x^{\tau_5}\sigma_x^{\omega_6}\ldots\sigma_x^{\tau_{a-1}}, & (\beta_1 = 1) \end{cases} \tag{H.2}$$

where all (σ_x-)observables acting on $\{\tau_i\}_{i=2}^{a-1}$ are associated with the measurements of Protocol `III` (i.e. the outcomes $\{o_i\}$) and all others are associated with Protocol `IV` (i.e. the outcomes $\{m_i\}$).

It is now our goal to show that all these measurement outcomes are uniformly random, and that there are no correlations between the measurement outcomes associated with any subset $S \subset Q$, with $Q = \{\omega_2, \tau_2, \ldots, \omega_{a-1}, \tau_{a-1}\}$ the set of qubits measured throughout both Protocol `III` and Protocol `IV`. Any such S has an associated observable

$$M_S = \bigotimes_{i\in S} \sigma_{b(i)}^i, \tag{H.3}$$

where $b(i) \in \{x, y\}$ indicates the type of support on qubit i as shown in Eq. (H.2). If M_S does not commute with at least one generator of the stabiliser (i.e. any operator from Eq. (H.1)), by Gottesman-Knill simulation, the measurement outcome for M_S is uniformly random 0 or 1. If this holds for *any* S, there cannot be any correlations between any of the measurement outcomes. The uniform randomness of the individual measurement outcomes follows readily for the case when S contains only a single qubit. We now show that any M_S indeed always anti-commutes with at least a single generator.

Suppose that M_S *does* commute with all generators but is non-trivial. If it has (non-trivial) support on τ_{a-1}, this is necessarily with σ_x. It will then not commute with $\sigma_z^{\tau_{a-1}}\sigma_z^{\omega_a}$ (the last generator of Eq. (H.1)) and hence cannot have support on τ_{a-1}. Then, if M_S has (non-trivial) support on ω_{a-1}, with either a σ_x or σ_y, it will not commute with the generator $\sigma_z^{\omega_{a-1}}\sigma_x^{\tau_{a-1}}\sigma_x^{\omega_a}$ - thus it cannot have support on ω_{a-1} either.

We can inductively go through the rest of the qubits in Q in reversed order, i.e. from right to left through the observable from Eq. (H.2). For $j \in \{a-2, a-3, \ldots, 3, 2\}$:

> Suppose M_S has non-trivial support on τ_j, it is of type σ_x. Since M_S has by construction no support on any qubit to the right of τ_j, it does not commute with the generator $\sigma_z^{\tau_j}\sigma_z^{\omega_{j+1}}$ - hence M_S cannot have support on τ_j.
>
> Suppose M_S has non-trivial support on ω_j, either of type σ_x or σ_y. Since M_S has by construction no support on any qubit to the right of ω_j, it does not commute with the generator $\sigma_z^{\omega_j}\sigma_x^{\tau_j}\sigma_x^{\omega_{j+1}}\sigma_z^{\tau_{j+1}}$ – hence M_S cannot have support on ω_j.

We conclude that there is no M_S with non-trivial support on at least a single qubit that does not anti-commute with at least one generator.

From this, we can conclude that there are no correlations possible between any set of measurement outcomes from $\{o_i\}$ and $\{m_i\}$, and that they are thus uniformly random and uncorrelated. Moreover, it stays uniformly random under any noise that does not add a bias in the used measurement bases (i.e. σ_x and σ_y).

H.1.2 | Dishonest participant

We are now allowing a single non-participant to deviate from the protocol in an arbitrary way. Let the index of this dishonest non-participant be i. To try to force any other node in the network to implicitly reveal their identity, i can actively perform a different measurement than described, where their outcomes would then be correlated with its (e.g.) direct neighbours. If these correlations then do not exist between their outcomes and the announced outcomes, then they can infer that these announced outcomes are artificial, and therefore that those who have announced them are in fact participants. Let this arbitrary measurement be represented by a 2-qubit POVM $\mu_i := \{\mu_i^j\}$, where without loss of generality $j \in \{1, 2, 3, 4\}$.

Slightly abusing notation by combining POVM elements and observables, the measurement operator then becomes

$$M = \begin{cases} \sigma_y^{\omega_2}\sigma_x^{\tau_2}\ldots\sigma_x^{\omega_{i-1}}\sigma_x^{\tau_{i-1}} \bigotimes \mu_i^j \bigotimes \sigma_x^{\omega_{i+1}}\sigma_x^{\tau_{i+1}}\sigma_y^{\omega_{i+2}}\ldots\sigma_x^{\tau_{a-1}}, & (\beta_1 = 0) \\ \sigma_x^{\omega_2}\sigma_x^{\tau_2}\ldots\sigma_y^{\omega_{i-1}}\sigma_x^{\tau_{i-1}} \bigotimes \mu_i^j \bigotimes \sigma_y^{\omega_{i+1}}\sigma_x^{\tau_{i+1}}\sigma_x^{\omega_{i+2}}\ldots\sigma_x^{\tau_{a-1}}, & (\beta_1 = 1). \end{cases} \tag{H.4}$$

Without loss of generality, the underlying network state is still the same[1] as in (H.1). Likewise, all of the single-qubit measurement operators in M for any node $j \neq i$ do not commute with at least one of these generators, indicating that the individual measurement outcomes are uniformly random 0 or 1.

Similar to before, the goal is to show that no choice of μ_i can create a measurement operator M_S that shows correlations between the qubits of i and any subset $S \subset Q$. It suffices to show that there is no M_S with support on any of the qubits in $Q \setminus \{\tau_i, \omega_i\}$ that commutes with all generators. By the same analysis as in the previous section, M_S cannot have any support on the qubits of any $j|j \in \{a-1, \ldots, i+1\}$. Moreover, we can make a similar inductive argument for nodes $j|j \in \{2, \ldots, i-1\}$. Independent of β_1, if M_S has support on ω_2 it will not commute with the generator $\sigma_x^{\tau_1}\sigma_z^{\omega_2}$. Likewise, if M_S has support on τ_2, it will not commute with the generator $\sigma_z^{\tau_1}\sigma_x^{\omega_2}\sigma_z^{\tau_2}$. We can inductively go through all qubits from the nodes $j|j \in \{2, \ldots, i-1\}$ to show that there exists no M_S that has non-trivial support on any qubit of the nodes $\{2, \ldots, i-1, i+1, \ldots, a-1\}$ and at the same time commutes with all the generators. We can conclude that, even for a dishonest node i, there are no correlations in the measurement outcomes announced by the other nodes.

[1] Any non-trivial map that i may perform on their subsystem can be merged with the measurements $\{\mu_i\}$. The other participants don't deviate, or i is not aware of the deviation and therefore cannot exploit it.