Practical Unconditionally Secure Signature Schemes and Related Protocols

Ryan Philip Amiri

Submitted for the degree of Doctor of Philosophy

Heriot-Watt University

School of Engineering and Physical Sciences

September 2017

The copyright in this thesis is owned by the author. Any quotation from the thesis or use of any of the information contained in it must acknowledge this thesis as the source of the quotation or information.

Abstract

The security guarantees provided by digital signatures are vital to many modern applications such as online banking, software distribution, emails and many more. Their ubiquity across digital communications arguably makes digital signatures one of the most important inventions in cryptography. Worryingly, all commonly used schemes – RSA, DSA and ECDSA – provide only computational security, and are rendered completely insecure by quantum computers. Motivated by this threat, this thesis focuses on unconditionally secure signature (USS) schemes – an information-theoretically secure analogue of digital signatures.

We present and analyse two new USS schemes. The first is a quantum USS scheme that is both information-theoretically secure and realisable with current technology. The scheme represents an improvement over all previous quantum USS schemes, which were always either realisable or had a full security proof, but not both. The second is an entirely classical USS scheme that uses minimal resources and is vastly more efficient than all previous schemes, to such an extent that it could potentially find real-world application. With the discovery of such an efficient classical USS scheme using only minimal resources, it is difficult to see what advantage quantum USS schemes may provide.

Lastly, we remain in the information-theoretic security setting and consider two quantum protocols closely related to USS schemes – oblivious transfer and quantum money. For oblivious transfer, we prove new lower bounds on the minimum achievable cheating probabilities in any 1-out-of-2 protocol. For quantum money, we present a scheme that is more efficient and error tolerant than all previous schemes. Additionally, we show that it can be implemented using a coherent source and lossy detectors, thereby allowing for the first experimental demonstration of quantum coin creation and verification.

Acknowledgements

The work contained in this PhD was carried out over the course of three years at Heriot-Watt University, Edinburgh, under the supervision of Erika Andersson, and as part of the Scottish Condensed Matter Physics Doctoral Training Centre. I begin by thanking Erika for the fantastic opportunities I gained as a student under her supervision, for providing the topic of my PhD, and for the academic freedom she allowed me in my research. I would also like to especially thank Petros Wallden, whose advice, guidance and supervision has been invaluable to me throughout my PhD. Without the many helpful discussions we've had on virtually all aspects of my work, this thesis would look very different.

I would like to thank my collaborators at Heriot-Watt University; in particular it was always a great pleasure to work with Gerald Buller, Robert Collins and Ross Donaldson. Not only were they great fun to work with, but also their experimental expertise and their patience in withstanding questions from a sadly obvious maths graduate helped me immensely. I am also grateful to all of my collaborators outside of Heriot-Watt and would especially like to thank Juan Miguel Arrazola, from whom I learnt a lot.

The organisers of the DTC through which I received funding also deserve a special mention for providing such a supportive, interesting and thoroughly enjoyable environment in which to work. The friends I gained as part of the DTC have enriched my PhD experience hugely. I will always look back fondly on the mobile war cabinet, precariously aiming for board game world domination through forceful diplomacy over a minibus table on the 8-hour journey from Oxford to Edinburgh. Max and Francisco, you betrayers, you should have sided with me...

Lastly, I would like to thank my parents, Denise and Iraj, my sister, Tianna, and my girlfriend, Toni, for pretending to read with interest the papers I published. Sincerely though, without their support none of this would have been possible.

Contents

List of research outputs

The following is a list of research outputs that have arisen as a result of the work contained in this thesis.

Publications

P1. Amiri, Ryan, and Erika Andersson. "Unconditionally secure quantum signatures." Entropy 17.8 (2015): 5635-5659.

P2. Donaldson, Ross J., Robert J. Collins, Klaudia Kleczkowska, Ryan Amiri, Petros Wallden, Vedran Dunjko, John Jeffers, Erika Andersson, and Gerald S. Buller. "Experimental demonstration of kilometer-range quantum digital signatures." Physical Review A 93, no. 1 (2016): 012329.

P3. Amiri, Ryan, Petros Wallden, Adrian Kent, and Erika Andersson. "Secure quantum signatures using insecure quantum channels." Physical Review A 93, no. 3 (2016): 032325.

P4. Collins, Robert J., Ryan Amiri, Mikio Fujiwara, Toshimori Honjo, Kaoru Shimizu, Kiyoshi Tamaki, Masahiro Takeoka, Erika Andersson, Gerald S. Buller, and Masahide Sasaki. "Experimental transmission of quantum digital signatures over 90 km of installed optical fiber using a differential phase shift quantum key distribution system." Optics Letters 41, no. 21 (2016): 4883-4886.

P5. Collins, Robert J., Ryan Amiri, Mikio Fujiwara, Toshimori Honjo, Kaoru Shimizu, Kiyoshi Tamaki, Masahiro Takeoka, Masahide Sasaki, Erika Andersson, and Gerald S. Buller. "Experimental demonstration of quantum digital signatures over 43 dB channel loss using differential phase shift quantum key distribution." Scientific Reports 7 (2017).

P6. Puthoor, Ittoop Vergheese, Ryan Amiri, Petros Wallden, Marcos Curty, and

Erika Andersson. "Measurement-device-independent quantum digital signatures." Physical Review A 94, no. 2 (2016): 022328.

P7. Amiri, Ryan, and Juan Miguel Arrazola. "Quantum money with nearly optimal error tolerance." Physical Review A 95, no. 6 (2017): 062334.

Submitted manuscripts

S1. Amiri, Ryan, Aysajan Abidin, Petros Wallden, and Erika Andersson. "Unconditionally secure signatures." Cryptology ePrint Archive, (Report 2016/739) (2016).

S2. Amiri, Ryan, Petros Wallden, and Erika Andersson. "Almost tight lower bounds for 1-out-of-2 quantum oblivious transfer." (2017).

Research visits

1. National Institute of Information and Communications, Tokyo. Research Internship in the Quantum Information and Communications Technology group, August 2015 - October 2015.

2. National University of Singapore, Singapore. Research visit to the Centre for Quantum Technologies, February 2017 - March 2017.

Funding awards

1. UK Science and Innovation Network grant, £1,500, Global Partnership Fund PSI 16004 SIN ASPAC UK-Japan. Award used to support travel to the UK-Japan Quantum Communication Workshop held on the 13th of October 2016 in Tokyo. Co-presented talk entitled "Quantum Digital Signatures".

2. SUPA Short Term Visitors grant, £1,500. Award used to support travel to further collaboration with researchers from the University of Science and Technology of China (USTC) and the Centre for Quantum Technologies (CQT), National University of Singapore. Delivered oral presentation entitled "Quantum money" at both USTC and CQT.

Patent applications

1. Patent application filed (application number GB1601759.2) for the "hash scheme" presented in Chapter 8.

Contributions

This thesis is the result of my own work carried out at Heriot-Watt University between September 2014 and August 2017. Parts of the work presented in this thesis have been published in refereed scientific journals. In all cases the text in the chapters has been written entirely by me. All figures have been produced by me unless explicitly stated in the text.

- Chapter 2 is a review of the existing USS literature. It is largely new, but takes elements from the review paper (P1), which was written jointly by myself and my supervisor, Erika Andersson (EA). Section 2.4 is based on Ref. [1], but I have updated some parts of the security analysis to reflect later work and discussions between myself, Petros Wallden (PW) and EA. Section 2.4.3 describes the quantum USS experiment published in (P2) for which I performed the security analysis, with guidance from PW and EA.

- Chapter 5 contains unpublished thoughts and results which arose as part of discussions between myself, PW, EA and Adrian Kent (AK). The considerations in that chapter served to guide the direction of research in later chapters. Section 5.4 is my own work.

- Chapter 6 is a modified and extended version of the paper (P3), written by me. It describes the AWKA scheme proposed jointly by me, PW, AK and EA. The security analysis and all other theoretical results contained in this chapter are mine, with supervision and guidance provided by PW, AK and EA. Section 6.6.2 describes the results of two experimental implementations of the AWKA scheme, published in (P4) and (P5). In collaboration with Masahiro Takeoka, Kiyoshi Tamaki and EA, I performed the security analysis and provided theoretical support for both implementations.

- Section 7.4 is based on the paper (P6), written by Ittoop Puthoor (IP). The research for this paper was performed in collaboration with IP, PW, Marcos Curty (MC) and EA. EA suggested the form of the protocol. The analysis of

the key generation protocol, the calculation of the min-entropy and the security analysis of the overall scheme was performed jointly by me, IP and MC. The simulation was performed by IP and MC. Although IP was responsible for writing (P6), the text in Section 7.4 is my own.

- Chapter 8 is an extended version of the paper (S1), written by me. It describes the hash scheme proposed jointly by me, Aysajan Abidin (AA), PW and EA. The security analysis in Section 8.3 was performed mainly by me, with help and guidance from AA and PW. All other results/considerations contained in this chapter are mine, with supervision from AA, PW and EA.

- Chapter 9 is taken from the paper (S2), written by me. The work in this chapter is entirely my own, with supervision from PW and EA.

- Chapter 10 is taken from the paper (P7), written by me. The research for this paper was performed in collaboration with Juan Miguel Arrazola (JMA). The paper is a joint work. JMA instigated the project and was active in leading the direction of the research. I was largely responsible for the details of the scheme and its security analysis, i.e. Sections 10.4 and 10.5. Section 10.6 was written jointly by myself and JMA.

Chapter 1

Introduction

Since its discovery in the early 20th century, quantum mechanics has been the fundamental theory used to describe nature. Though strange and counterintuitive, the postulates of quantum mechanics have been tremendously successful in describing physical systems and their evolution, despite the fact that a true understanding of the meaning of these postulates remains elusive. Perhaps the most striking feature of quantum mechanics is the uncertainty principle, which places absolute limits on the precision with which we can measure two non-commuting observables. Originally formulated by Heisenberg in 1927 [2], and extended by Robertson in 1929 to the now well-known form [3], the uncertainty principle has profound consequences for nature. Together with entanglement theory, it has sparked many of the most well-known and controversial debates in the history of quantum mechanics [4, 5]. On the face of it, the uncertainty principle seems to be an inherently negative result, severely limiting our ability to accurately resolve certain observables. Nevertheless, physicists are nothing if not resilient. What was originally considered to be a strict limitation has since been transformed into a useful cryptographic resource, becoming the cornerstone of the exciting field of quantum cryptography.

In the late 1940's the mathematical foundations of information theory and cryptography were established by Shannon in his seminal papers [6, 7]. Though seemingly separate from physics, Shannon's insight was to define and introduce the concept of information entropy to quantify uncertainty[1]. With the progression of computing technologies, information theory and cryptography flourished, proving themselves essential for understanding core concepts in the communication, storage and manipulation of data. Traditionally, cryptography has been associated solely with keeping

[1]The similarity to thermodynamic entropy was immediately apparent, revealing an intrinsic link between the notions of physics and information, though the underlying reason for the connection was not, and still is not, fully understood.

information secret. In reality, the explosion of communication and computation over the last few decades has seen the field of cryptography grow enormously to encompass many different tasks and protocols.

Of central importance to this thesis is the cryptographic notion of a signature scheme, first proposed in 1976 as a means of safeguarding the integrity, authenticity and transferability of a message [8]. Signatures guarantee the identity of the sender and ensure that the contents of a message have not been modified in transit. Importantly, they do so in a verifiable way so that participants can be held accountable to anything they have signed, or can prove when a document has been forged. Digital signatures have since become ubiquitous across modern communications, finding applications in online banking, software distribution, emails, legal documents, photography and many more. Their widespread applicability has led to them being described as "one of the most important inventions in modern cryptography" [9].

With the benefit of hindsight, it is perhaps surprising that the inherent suitability of quantum mechanics to cryptography remained largely overlooked for more than 30 years until the discovery of quantum key distribution (QKD) [10] in 1984[2]. QKD harnessed the power of quantum mechanics to achieve something that is provably impossible to do in the classical world – distribute a secret key between two parties with information-theoretic security[3]. The discovery highlighted the vast practical potential that quantum mechanical effects have in cryptography, and founded the field which is today called quantum cryptography. In the last 30 years, theoretical and technological advances have seen this already rich field mature and expand at an exponential rate. Nevertheless, despite their importance, signatures have until recently remained relatively untouched by the quantum cryptography community. In this thesis we will explore signatures from the viewpoint of a quantum physicist, and try to discover what advantages uniquely quantum effects may provide.

1.1 Motivation

The most common signature schemes in use today are public-key schemes based on the Rivest–Shamir–Adleman (RSA) algorithm [12], the Digital Signature Algorithm (DSA) [13], and the Elliptic Curve Digital Signature Algorithm (ECDSA)[14]. These schemes are believed to provide *computational security*, which means that there is

[2]Though pre-dated by Wiesner's unpublished 1970 "Conjugate Coding" paper [11].

[3]In this thesis we will use the phrases "unconditional security" and "information-theoretic security" interchangeably. If a protocol is called unconditionally secure and no conditions are explicitly stated, then it means that within the assumptions of the protocol it has been proven secure against all types of attack allowed by quantum mechanics.

no probabilistic polynomial-time algorithm that can solve the underlying problems upon which the scheme is built. In practice, this means that the schemes remain secure assuming the adversary has bounded computational resources. In 1994, the cryptography community was shocked by the discovery of a polynomial-time *quantum* algorithm that solved both the factoring and discrete logarithm problem [15], effectively rendering RSA, DSA and ECDSA completely insecure in the presence of a quantum computer. While the threat from quantum computers may as yet seem remote, in many cases it is necessary to keep data secure for years or decades. Furthermore, government and corporate infrastructures can be hugely complex so that structural changes take many years to implement. In security, preparation and foresight are key; if one is truly concerned about long-term security, then one must protect against both current and future threats. Indeed, in response to these future threats, in August 2015 the National Security Agency in the USA recommended a transition to post-quantum secure algorithms, hailing in the beginning of the era of post-quantum security.

Broadly speaking, post-quantum secure algorithms fall into two categories:

1. **Unconditionally secure algorithms.** These provide the highest level of security – security that holds regardless of the computational resources available to an adversary.

2. **Quantum-safe algorithms.** These provide computational security against a quantum adversary, i.e. an adversary able to implement quantum algorithms.

This thesis is concerned with signature schemes in the first category, called unconditionally secure signature (USS) schemes, and addresses some of the many open questions in this young and largely unexplored field, such as:

- Can we construct efficient USS schemes?

- What resources/assumptions are necessary for USS schemes to be possible?

- Does quantum mechanics allow for more efficient USS schemes? Does it allow for schemes requiring fewer resources?

- Does the ability to perform USS schemes imply the ability to perform other cryptographic protocols?

- What is the relationship between USS schemes and other cryptographic protocols such as QKD, Byzantine agreement and oblivious transfer?

1.2 Thesis outline

In Chapter 2 we will introduce post-quantum signature schemes in more detail, with an emphasis on USS schemes. We review the existing work in this field and finish the chapter with an in-depth look at the "state-of-the-art" quantum USS signature scheme which formed the starting point of the work in this thesis.

In Chapter 3 we provide an overview of some important definitions and results in quantum cryptography that will be necessary to understand the results contained in this thesis. Additionally, we introduce some useful concepts from classical cryptography including message authentication codes, Byzantine agreement and oblivious transfer.

In Chapter 4 we provide the formal security framework for quantum USS schemes.

In Chapter 5 we describe some simple attacks and resource considerations one must take into account when creating practical USS schemes. We use these to motivate the scheme presented and analysed in the following chapter.

In Chapter 6 we leverage modern techniques from QKD and apply them to signatures. The result is the first practical quantum USS scheme with a full security proof, and which does not rely on any undesirable assumptions that have been present in all previous quantum schemes, such as "tamper-proof" quantum channels or long-term quantum memory. As well as providing a rigorous security analysis, we are also able to use the scheme to find interesting differences in resource requirements between quantum USS schemes and QKD. Lastly, we mention various experimental implementations of the scheme and comment on their efficiency.

In Chapter 7 we extend the protocol of the previous chapter to make it measurement-device-independent (MDI). By removing side channels that are commonly hacked, MDI schemes help to bridge the gap between the theory and real-world implementations, at the cost of a significant reduction in signing efficiency. The scheme also enjoys some secondary benefits such as an increased transmission distance and a possible reduction in cost.

In Chapter 8 we explore in more depth the question of exactly how quantum mechanics can help to create USS schemes. We present a classical USS scheme using only minimal resources and assumptions which, compared to all previous USS schemes (both classical and quantum), is able to drastically increase efficiency while also maintaining security.

In Chapter 9 we consider 1-out-of-2 oblivious transfer (1-2 OT) schemes in the information-theoretic security setting. Perfect 1-2 OT is known to be impossible in

this setting, but it is possible to devise schemes in which the participants abilities to cheat are restricted. We prove new lower bounds on the cheating probabilities that must inevitably arise in any 1-2 OT protocol, and use these bounds to gain insight into the potential use of imperfect 1-2 OT schemes in other cryptographic protocols, including USS schemes.

In Chapter 10 we consider secret-key quantum money schemes with classical verification in the information-theoretic security setting. We describe and analyse a new scheme with a number of benefits over all previous proposals. We further prove bounds on the maximum noise tolerance possible for a wide class of quantum money schemes. Lastly, we show that our scheme can be mapped to one using a coherent state source together with lossy and imperfect detectors. The resulting scheme remains secure while also allowing, for the first time, for coins to be created and verified with current technology.

Chapter 2

Post-quantum signatures

2.1 Introduction

At a high level, signature schemes are often viewed primarily as a method by which digital communications can be authenticated, i.e. they allow the recipient of a message to deduce whether the contents of the message have been altered in transit. However, they also provide other important guarantees. These are:

1. Non-forging: signatures can be used to authenticate not only the contents of the message, but also the source of the message.

2. Universal verifiability/transferability: if a recipient accepts a signed message from a source S, then she can be sure that any third party would also be able to verify for themselves that the message is valid and originated with S.

3. Non-repudiation: a sender cannot send a signed message, and later deny having done so.

As mentioned in the previous chapter, there are two main classes of signature scheme providing security in a post-quantum world: quantum-safe digital signature schemes and USS schemes. The difference between the two classes lies in the level of security they provide: quantum-safe schemes provide computational security whereas USS schemes provide information-theoretic security. This difference in security level leads to some subtle differences in protocol functionality, and as such security guarantees specific to signatures in the information theoretic-security setting are provided in Section 2.3.2.

Quantum-safe signature schemes have been widely studied, and we begin this chapter by providing an overview of the research in that area. USS schemes on the other hand have been investigated less, and are the focus of this thesis. Starting

in Section 2.3, in this chapter we provide an in-depth introduction to USS schemes and review the existing research in this field.

2.2 Quantum-safe signature schemes

Quantum-safe signatures are cryptosystems that are not yet known to be vulnerable to quantum adversaries with bounded computational resources. In practice, this means that there is no known polynomial-time quantum algorithm that breaks the security of the schemes. Many of these schemes are quite new and have not yet stood the test of time, but are nevertheless widely expected to be hard to solve, even for quantum adversaries. It should be stressed that, even if the underlying problems are proved to be hard, the security provided would still be computational, and the systems would still be vulnerable to brute-force attacks.

Quantum-safe signature schemes have the important advantage of being public-key schemes, meaning they have the "universal verifiability" property inherent to standard digital signatures[1]. Below we summarise some of the most promising quantum-safe encryption/signature schemes. Note that any public-key encryption scheme can be used to create a public-key signature scheme. For example, a simple way to do this is to apply a Fiat-Shamir transformation to a public-key identification protocol [16].

Lattice-based cryptography

Lattice-based signature schemes are arguably the most promising and well studied class of quantum-safe schemes. In its simplest form, a mathematical lattice of integers is just a discrete subgroup of the additive group $\mathbb{Z}^n$. The fundamental problem upon which lattice-based cryptography is founded is the problem of finding the shortest non-zero vector within the lattice. It is called the Shortest Vector Problem (SVP), and it is known to be NP-hard. However, as for many commercial schemes, efficiency trumps security, and all practical systems are based on weaker variants of the SVP, for which the computational difficulty is unknown. Two common families of lattice-based cryptosystems based on weaker variants of the SVP are: Learning With Error (LWE) schemes, such as Ref. [17]; and SS-NTRU schemes, such as Ref. [18]. The security of both of these families of schemes can be reduced to the same lattice problem.

[1]Digital signatures is a term often reserved for schemes exhibiting exactly this property. As such, in this thesis we refrain from describing schemes as a *digital* signature scheme unless it provides universal verifiability.

Lattice-based schemes have found popularity due to their short signature size and the relative computational ease of generating a signature. However, it is worth noting that practical lattice schemes have not yet stood the test of time, and many believe that the LWE problem is vulnerable to quantum attacks similar to Shor's algorithm[2]. Still, as of the present day, there is no known efficient algorithm solving the LWE problem, and lattice-based schemes are considered by many to be the most likely quantum-safe successor to the existing digital signature schemes. Currently, the most efficient lattice-based signature scheme is BLISS [19].

Multivariate cryptography

Multivariate cryptography is the term given to cryptosystems where the trapdoor one-way function is a multivariate quadratic polynomial map over a finite field. The public key is usually given by a set of multivariate polynomials, and encryption involves evaluating these polynomials with the message given as input. Decryption involves inverting the multivariate quadratic map, a problem which can be shown to be NP-hard [20]. Of course, to allow for a trapdoor one must provide additional structure to the polynomials chosen as a public key. This means that the inversion problem is no longer necessarily NP-hard, but only believed to be hard.

Nevertheless, multivariate cryptosystems are widely believed to remain secure even in the presence of quantum computers. Despite this, multivariate cryptosystems have not seen widespread use since they suffer from large public and private key sizes, and are relatively computationally expensive to use. On the other hand, the required signature length is very small, meaning there may be applications for which multivariate schemes are preferable. The most popular multivariate signature schemes are the Unbalanced Oil and Vinegar scheme (UOV) [21] and the Rainbow scheme [22].

Code-based cryptography

Code-based cryptosystems were first proposed in 1978 by McEliece [23]. They use efficient error correcting codes, such as Goppa codes [24], to scramble and decode messages. The underlying problem upon which security is based is that of decoding a general linear code, also called syndrome decoding, which is known to be NP-hard.

[2]Indeed, in November 2016 a paper authored by Lior Eldar and Peter Shor appeared which claimed to make a significant breakthrough in the search for an efficient algorithm to solve LWE. However, the paper was later retracted due to a mistake which invalidated the result. Nevertheless, the work highlights that there is significant doubt regarding whether LWE lattice schemes are truly secure.

The private key is an error-correcting code which can efficiently correct up to t errors. The public key is a random generator matrix of a randomly permuted version of the private key. Encryption is performed by adding t errors to the message, where the errors are chosen using the public key. Only the holder of the private key is able to remove the errors generated by the public key.

The security reductions of code-based cryptosystems are well understood and believed to be strong. Further, the McEliece scheme is very fast to use, as the computational complexity of both encryption and decryption is low. However, McEliece schemes have never become popular, mainly due to the large size of their public key relative to competing schemes – often a public-key of several megabytes is necessary for 128-bit security.

Hash-based cryptography

Hash-based signatures are created using any cryptographic hash function. A cryptographic hash function is a hash function, h, exhibiting the following properties:

1. Pre-image resistance: Given $h(x)$, it should be difficult to find x.
2. Second pre-image resistance: Given x_1, it should be difficult to find x_2 such that $h(x_1) = h(x_2)$.
3. Collision resistance: It should be difficult to find any distinct pair x_1 and x_2 such that $h(x_1) = h(x_2)$.

The existence of a digital signature scheme that can sign multiple messages using a single private key implies the existence of a hash-based signature scheme. Therefore, cryptographic hash functions are a minimal requirement for the existence of *any* secure digital signature scheme that can sign more than one message using a single private key [20]. In this sense, hash-based schemes are the most important and fundamental digital signature scheme.

The security of hash-based signatures relies solely on the collision resistance of the underlying cryptographic hash function. This means that hash-based schemes are extremely adaptable – if the underlying hash function is found to be insecure, then it can simply be switched for another without overhauling the security systems in place. This flexibility makes hash-based schemes good candidates for providing post quantum security in an uncertain future [20].

Hash-based signatures are widely modelled on the Lamport-Diffe one-time signature scheme, which can be implemented using any one-way function f. To sign

an n-bit message, the private key is a collection of $2n$ randomly chosen bit strings, two for each of the bits contained in the message. The public key is the collection of hash tags generated from applying f to each of the strings in the private key. To sign a message, the sender attaches the n private keys corresponding to each of the bit values in the message. Since the private key is partially revealed, it cannot be reused, hence the name "one-time" (see Section 2.3.3 for a more in-depth discussion).

Finite reusability can be enforced by augmenting a one-time scheme with structures such as binary trees. This idea was first introduced by Merkle [25], but the construction suffered many efficiency drawbacks such as large public and private keys, long signature lengths and computationally intensive signature generation. An additional disadvantage associated with finite reusability is that each use of a Merkle-type hash-based scheme is not independent meaning one must keep track of the *state* of the algorithm, i.e. which one-time keys have been used and the position of the algorithm on the binary tree. In large-scale environments, statefullness is difficult to manage [26]. Nevertheless, the techniques have improved over the years and one of the current best schemes, XMSS [27], is less computationally intensive and requires much smaller keys, though it is still stateful and the signature length is a factor of 10 larger than schemes such as RSA. Stateless hash-based schemes also exist [28], and this field is an active area of research.

2.3 Unconditionally secure signatures

For real-world applications, two highly desirable properties of digital signatures are the fact that they are non-interactive (message recipients do not need to communicate for the message to be transferable), and they do not require participants to share an initial secret key. In practice, this means that digital signature schemes exhibit the "universal verifiability" property. To gain universal verifiability digital signatures sacrifices security, and all schemes with this property are only computationally secure. USS schemes do not provide universal verifiability, but instead use a set-up phase to ensure that messages are transferable to any recipient that was part of the set-up phase.

Nevertheless, for particularly high-security applications it may be desirable to use schemes providing unconditional security. USS schemes do just that, and are provably secure even when the time and computational resources available to the adversary are completely unknown. Unfortunately, such a high level of security carries a cost, and to gain unconditional security one must sacrifice the universal

verifiability property inherent to all currently used digital signature schemes. Instead, USS schemes always contain at least two stages: a *distribution phase* in which the protocol is set up; and a *messaging phase* in which the message is actually sent. Only those participants who took part in the distribution stage can later send and receive signed messages. Moreover, we prove in Theorem 5.1 of Chapter 5 that the distribution stage of all USS schemes is necessarily interactive (i.e. potential message recipients must all be able to communicate either directly or indirectly with all other recipients) and always requires a secret shared key between all participants. This caveat means that USS schemes are functionally different to traditional digital signatures, but maintain very similar goals. The interactive nature of USS schemes, together with their requirement of shared secrets, exclude them from being a realistic replacement for many core applications of digital signatures, but does not rule out their use in some high-security scenarios.

2.3.1 Generic USS scheme

Unlike many cryptographic primitives, USS schemes do not assume that any party is honest. Instead, it is typically assumed that more than a threshold number of participants are honest, but that the identities of the dishonest participants are unknown to all. USS schemes are divided into two categories: quantum schemes, whose security is derived from physical laws; and classical schemes, whose security is derived from mathematical reasoning. Regardless of this, all USS schemes have the same basic two-stage structure:

- **The distribution stage:** this is the set-up phase in which a finite number of participants interactively communicate in order to distribute information that will later be used to sign and/or verify messages. All participants who may want to send/receive a signed message in future must take part in this stage. This stage can involve quantum communication, in which case we call the scheme a quantum USS scheme, or only classical communication, in which case we call the scheme a classical USS scheme.

- **The messaging stage:** this is the phase in which the message is sent. This stage can happen at any point in time after the distribution stage, and for signatures to be useful it must be non-interactive (i.e. communication is only required between the sender and the receiver of the message). The sender (normally called Alice) sends the message together with the signature, set up in the distribution stage, to the desired recipient (normally called Bob). Bob

should be able to check the validity of the signed message locally, using no interaction with other participants. If Bob wants to transfer the message to a third party (normally called Charlie), then Bob simply forwards the message-signature pair to Charlie, who can again check the validity of the pair non-interactively.

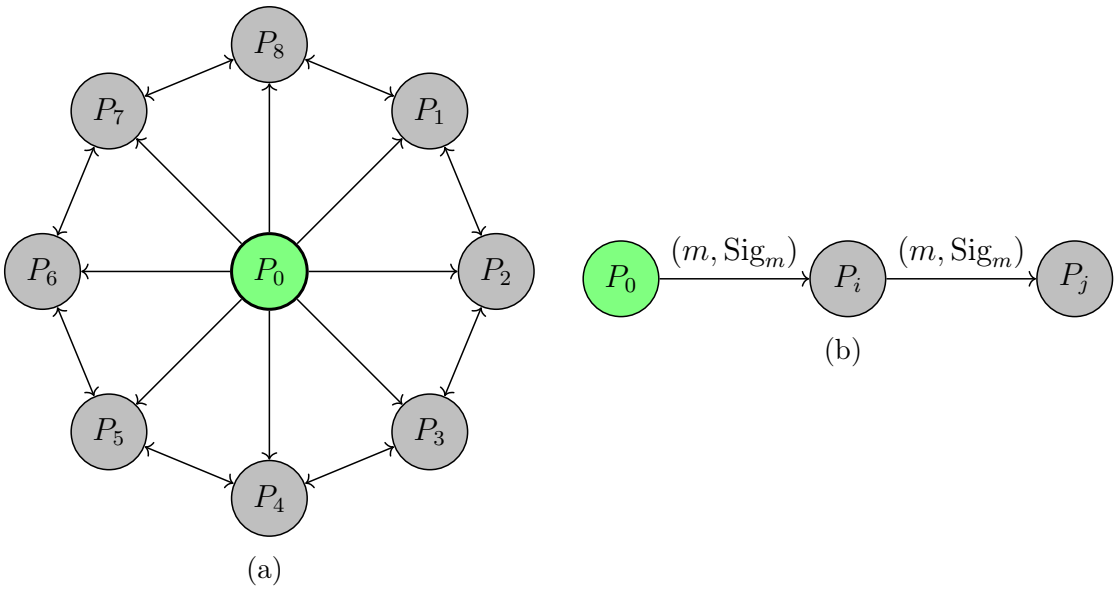


Figure 2.1: The figure shows the two stages – the distribution stage and the messaging stage – of a generic USS scheme. Figure (a) shows the generic distribution stage for a sender and $N = 8$ recipients. Note that communication is restricted to nearest neighbour for illustration purposes alone. In reality, recipients will normally communicate pairwise with *all* other recipients. Communication could be quantum or classical depending on the protocol. Figure (b) shows the generic messaging stage showing a message-signature pair, (m, Sig_m), sent to recipient P_i and subsequently transferred to recipient P_j. The message m is always a classical bit string, and is guaranteed to be transferable *in sequence* a fixed number of times (specified by the protocol).

It is important to stress that all USS schemes considered in this thesis are designed to secure *classical* information, and we do not consider the various schemes proposed to secure *quantum* information [29, 30]. When we refer to a "quantum" or "classical" USS scheme, the distinction applies only to the resources required in the distribution stage, not the information being signed. Although the resources available may be highly application dependent, in many cases it will be desirable to build schemes which make minimal resource assumptions. As such, we define the standard resource model which contains the resources most commonly assumed throughout this thesis.

Definition 2.1 (The Standard Resource Model)**.** For classical USS schemes, we assume that all participants are connected pairwise by both authenticated classical channels and secret classical channels. For quantum USS schemes we assume that participants are connected pairwise by both authenticated classical channels and insecure quantum channels.

Note that there is no physical difference between an authenticated classical channel and a secret classical channel – any insecure channel that can be used to transmit classical information can be transformed into either an authenticated or a secret channel without changing the channel itself, only the inputs to the channel. In the information-theoretic setting, the difference between an authenticated classical channel and a secret classical channel is the amount of secret shared key required. To authenticate an n-bit message, the sender and receiver must share $O(\log n)$ bits of secret key [31]. This key is used to append an authentication tag onto the message which is inputted into the communication channel. To send the message in secret the sender and receiver must share $O(n)$ bits of secret key [7]. This key is used to one-time-pad the input into the communication channel so that the ciphertext is transmitted rather than the plaintext message. Concretely, for an n-bit message m and an n-bit key k, the one-time-pad simply outputs ciphertext $m \oplus k$ [32]. Therefore, although it is often convenient to talk about resources in terms of authenticated/secret channels, in some scenarios it is clearer to talk only about secret-bit requirements, since these can be used to generate both authenticated *and* secret channels.

Note also that the different resources assumed in classical and quantum schemes (secret classical channel vs insecure quantum channel) is in some sense only a matter of perspective. This is because, in the information-theoretic setting, creating a secret classical channel requires a secret shared key. A secret shared key can only be generated between two parties (separated by an unsecured distance in space) in a provably secure way via QKD, which in turn requires an insecure quantum channel.

2.3.2 Security requirements of USS schemes

Recall that for USS schemes any participant could be dishonest. The motivations, powers and strategies available to the adversary are highly dependent on the adversary's identity; for example, an adversarial coalition including the sender will never try to forge a message, but will often have more power than a coalition not including the sender. Therefore, when considering what it means for the protocol to be secure, one must classify the different powers and attacks available to the adversary given that the adversary could be any subset of the participants. Since the identities of the dishonest participants are unknown, secure protocols must protect against all types of dishonest behaviour simultaneously. Informally, security in the information-theoretic setting means that the signature scheme has the following three properties [33]:

1. Unforgeability: Except with negligible probability, it should not be possible for an adversary to create a valid signature.

2. Transferability: If a verifier accepts a signature, he should be confident that any other verifier would also accept the signature.

3. Non-repudiation: Except with negligible probability, a signer should be unable to repudiate a legitimate signature that she has created.

Formal security definitions for USS schemes are provided in Chapter 4 [34], and similar definitions for classical USS schemes can be found in [33].

Since transferability is a security requirement of all signature schemes, the minimum number of participants in any USS scheme is three. USS schemes are always unable to handle more than one half of the participants being dishonest (see Chapter 4) without introducing additional trust assumptions such as a trusted authority, something which is avoided in this thesis. Therefore, the simplest scenario to consider is the three participant case in which at most one participant (whose identity is unknown) is dishonest.

The development of practical USS schemes has progressed incrementally, with new protocols being proposed to address specific issues in previous protocols. In the following sections, we provide an overview of the most important such schemes in order to motivate the design and development of later schemes.

2.3.3 Lamport-Diffie one-time signatures

An extremely useful resource in the development of practical cryptography is the notion of a one-way function. Informally, this is a function whose output is easy to compute given an input, but whose input is computationally difficult to compute given an output[3]. A standard example is prime factorisation: given two large prime numbers it is easy to compute their product, but given their product it is difficult to find the prime factors. This asymmetric nature is the foundation of public-key cryptography, and has been immensely useful in generating efficient cryptosystems. Signatures are closely related to one-way functions, and indeed it has been shown that one-way functions are necessary and sufficient for (computationally) secure signatures [35].

[3]It is not known whether one-way functions actually exist. Despite there being many functions which seem to be one-way, actually proving this is the case would imply $P \neq NP$, and as such all currently used functions are only *believed* to be one-way.

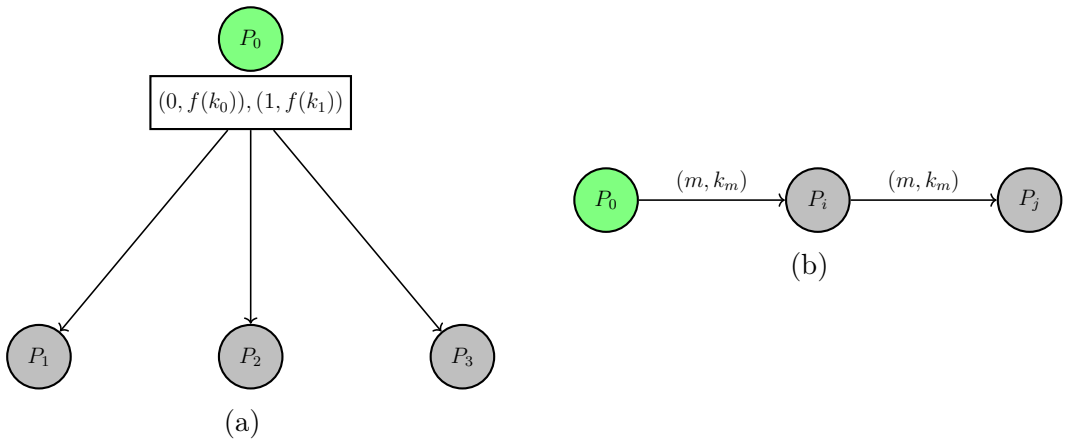


Figure 2.2: The figure gives a simplified schematic of how the Lamport scheme can be used to sign a one-bit message. In Figure (a), for all possible future messages m, Alice randomly picks k_m and computes $f(k_m)$. She broadcasts $(m, f(k_m))$ to all possible recipients. In Figure (b), to sign message m, Alice sends it together with the secret key, k_m, to the desired recipient. Of course, since a secret key is now known, the scheme is not reusable.

Lamport signatures [36] are a particularly simple class of one-time hash-based signature schemes which can be (computationally) securely implemented using any one-way function. Although Lamport signatures provide only computational security, many USS schemes are heavily based on the structure of Lamport signature schemes, but use additional resources to create an information-theoretic analogue of a one-way function, thereby generating unconditional security (see Section 2.3.5). For this reason it is useful to provide a brief illustration of a Lamport signature.

Lamport scheme

Imagine that Alice wants to send a single signed bit, 0 or 1, at some point in the future. In the distribution stage, Alice randomly chooses two inputs, k_0 and k_1, to the publicly known collision-resistant one-way function, f. She computes $f(k_0)$ and $f(k_1)$ and broadcasts[4] the outputs, $\{(0, f(k_0)), (1, f(k_1))\}$, as her public key. Since the function is assumed to be one-way, potential forgers cannot find an input generating either $f(k_0)$ or $f(k_1)$. In the messaging stage, to sign message m, Alice would send (m, k_m). The recipient would apply the publicly known f to k_m and accept the message only if $f(k_m)$ matches the public key. Clearly, if the participant transferred the message-signature pair to a third party, they would also find it to be valid. Once the message is sent, the public key cannot be re-used and must be discarded, hence the name "one-time" signatures.

[4]The use of a broadcast channel is not strictly necessary, but simplifies the protocol statement.

2.3.4 Classical USS schemes

The field of classical USS schemes has received relatively little attention. This is largely due to public-key digital signatures being vastly more practical, both in terms of efficiency and functionality, and therefore better suited to most applications. Nevertheless, there may still be situations involving highly sensitive information in which USS schemes are desirable due to the higher level of security they provide. Examples might include high-value banking transactions, signing important legal documents, or securing sensitive government communications. From a purely theoretical viewpoint, the question of "what are the advantages and limitations of signature schemes providing unconditional security?" is also interesting in its own right.

Existing schemes

The original classical USS scheme, proposed in 1988 by Chaum and Roijakkers [37], had the same one-time structure as the Lamport signature outlined above. The distribution stage required an authenticated broadcast channel as well as pairwise secret authenticated channels between all participants. For each possible future message that Alice could send, the participants make use of the untraceable sending protocol [38] to send her a string of secret bits anonymously. In the messaging stage, Alice's signature for message m is simply composed of all of the secret bit values that she received associated to m. Intuitively, security against forging is guaranteed because all participants send their elements of the signature over secret channels so that no one, except the sender, can reproduce the full signature. Transferability is limited to a single transfer in this scheme and is guaranteed by the anonymous channels, which means Alice is unable to bias a signature so that one party is more likely to accept than another. The same arguments show that Alice cannot repudiate a valid message.

Under the name of pseudosignatures, Pfitzmann and Waidner [39] generalised the above scheme to make it significantly more efficient, as well as finitely transferable. The recipients in this scheme use authenticated broadcast channels together with the untraceable sending protocol to anonymously transmit universal hash functions, rather than bit strings. In the messaging stage of the protocol, the hash functions are applied to an arbitrary (but size-bounded) message and appended as a tag. This change has the significant efficiency advantage of allowing longer messages to be sent using a *single* set-up phase, but still suffers from the one-time use restriction and requires expensive resources such as authenticated broadcast and anonymous

channels.

Other notable classical USS schemes include the variants by Hanaoka et al. [40, 41], constructed using polynomials over finite fields. These schemes require a trusted authority as well as secret channels. The inclusion of a trusted authority greatly simplifies both the set-up phase and the security proofs for transferability. Importantly, the Hanaoka schemes are not one-time – they are finitely re-usable meaning that the distribution phase does not have to be repeated for each message being sent. Nevertheless, the length of both the signature and the secret keys needed to generate signing/verification algorithms are still rather long, severely limiting its use in practice. A later variation of this scheme was proposed by Hanaoka *et al.* in [42]. This scheme sacrificed the re-usability of the previous scheme to achieve a reduction in the size of the secret keys needed to generate signing/verification algorithms by approximately a factor of 10.

There is also protocol P2 [1], originally introduced as a quantum USS scheme, but more properly classified as a classical scheme. For this protocol, in the three-party setting, Alice holds two secret keys for each possible future message she could send. In the distribution stage, for each possible future message she uses secret classical channels to send one of her secret keys to Bob and one to Charlie. Bob and Charlie then use a secret classical channel to exchange half of the bits they received from Alice. In the messaging stage, Alice's signature for message m is the two keys associated with that message (one of which was sent to Bob, and one of which was sent to Charlie). Though this simple scheme is less efficient than some of the schemes above, it shows that classical USS schemes exist which use only the resources available in the standard resource model.

Lastly, Ref. [43] considers classical USS schemes from an information-theoretic achievability perspective, and provides a full characterisation of the initial correlations required for signatures to be possible between three parties. This effectively quantifies the aims of any generic distribution stage in the case of just three parties.

2.3.5 Quantum USS schemes

As discussed above, one-way functions are extremely useful in cryptography for providing computational security. If we want unconditional security, is there a notion of an unconditionally secure one-way function? Classically the answer is no, since a brute force search will always yield all inputs leading to a given output. However, if we allow the output of the function to be a quantum state, then the answer is yes

– unconditionally secure quantum one-way functions do exist[5].

The one-way function we will consider was first introduced in the context of quantum fingerprinting [45, 46], and maps a b-bit classical string, s, to a quantum state of dimension d according to

$$s \to |\psi_s\rangle = \frac{1}{\sqrt{d}} \sum_{i=1}^{d} (-1)^{E(s)_i} |i\rangle. \tag{2.1}$$

Here, E is an error-correcting code mapping b-bit strings to d-bit strings, where $d = cb$ for some integer $c > 1$. The right-hand side can be viewed as a state containing $q = \log(cb)$ qubits. The one-way property follows from the Holevo bound [47, 48]:

Theorem 2.2 (The Holevo bound)**.** *Suppose Alice prepares a state ρ_X, where $X = 0, \ldots, n$ with probabilities $p_0, \ldots, p_n$. Bob performs a measurement described by POVM[6] elements $\{E_y\} = \{E_0, \ldots, E_m\}$ on that state, with measurement outcome Y. The Holevo bound states that for any such measurement Bob may do:*

$$I(X : Y) \leq S(\rho) - \sum_x p_x S(\rho_x) \tag{2.2}$$

where I is the mutual information, S is the von Neumann entropy and $\rho = \sum_x p_x \rho_x$.

Suppose that each ρ_x in the ensemble is a q-qubit state. A simple corollary of this theorem can be proved by noting that ρ is also a q-qubit state, meaning $S(\rho) \leq q$ and it is therefore impossible to retrieve more than q bits of information from a q-qubit state. In relation to the one-way function above, this corollary means that given access only to the output quantum state, it is impossible to derive the b-bit input as long as $b > \log d$.

The Gottesman-Chuang quantum USS scheme

The study of quantum USS schemes began with a paper by Gottesman and Chuang [49], in which the authors outline a Lamport-type "public key" quantum USS scheme relying on the information-theoretic one-way function described in Eq. (2.1).

To set up the scheme for a 1-bit message, Alice randomly chooses her private key to be two b-bit classical strings (k_0, k_1). To generate the public key, Alice applies the

[5]This is not to be confused with the quantum one-way functions defined in [44] as a function that is easily computable by a classical algorithm, but computationally hard to invert even by a quantum computer.

[6]POVM's are defined in Section 3.2.

one-way mapping (2.1) to create the two q-qubit states $(|\phi_{k_0}\rangle, |\phi_{k_1}\rangle)$. The mapping is known to all protocol participants, but the strings (k_0, k_1) are not. Alice sends t copies of the public key to each recipient in the scheme. For security, it is necessary to in some way authenticate these transmissions. The authors suggest using either a trusted distribution centre or authenticated quantum channels[7]. The participants then use $t-1$ of their copies to perform distributed swap tests to ensure they each received the same public key.

In the messaging stage, when Alice wants to send a signed message, m, she sends the pair (m, k_m) to the chosen recipient, say Bob. With the identity of k_m revealed, Bob can apply the known mapping to create as many copies of $|\phi_{k_m}\rangle$ as he likes, and can compare them with the public key. The comparison could be implemented, for example, using a swap test. If the created states match the public key, the message is accepted, else, it is rejected. To forward the message, Bob simply forwards on (m, k_m).

Security against non-transferability and repudiation is guaranteed by the symmetry enforced by the distributed swap tests performed in the set-up phase. To provide security against forging, the number of recipients, N, must be limited so that $Ntq \ll b$. This requirement, together with the Holevo bound, ensures that no adversarial coalition can derive the private key even with access to all copies of the public key. Longer messages can be signed by applying the above scheme bit-by-bit to each bit of the message.

Limiting factors

In terms of efficiency and practicality of the Gottesman-Chuang scheme, there are a few observations to make regarding its limiting factors. Consideration of these drawbacks has motivated later work on quantum USS schemes.

(I) The fingerprinting states (2.1) suggested for use as the public keys are highly entangled states, which would be experimentally difficult (or impossible) to create with current technology.

(II) The protocol requires long-term quantum memory to store the public key from the time the participants receive it, to the time (arbitrarily far in the future) when the sender wishes to transmit a signed message.

[7]Contrary to classical channels which can be authenticated but not encrypted, for quantum channels authentication implies encryption [29]. Therefore, authenticated quantum channels are considered to be an expensive resource.

(III) Since messages are signed bit-by-bit, the signature length scales linearly with the length of the message. Considering that the signature must be appended to the message, this scaling is highly inefficient[8], and must be improved before any protocol can be called practical.

(IV) The authors assume authenticated quantum channels which can be difficult to realise in practice. It would be useful to find methods of relaxing this assumption.

(V) When proving security against non-repudiation and transferability, the authors make the simplifying assumption of there being only three participants in the scheme. Additional participants add significant layers of complexity when considering possible cheating strategies. A full analysis of these strategies would be desirable.

(VI) This is a one-time signature scheme, meaning that the public/private key pair can only be used once. It would be highly desirable to find some method of making the keys re-usable.

Subsequent work

In light of the above efficiency and practicality limitations, a series of papers successively improved upon this original and provided solutions to problems (I) and (II). The first progress was made by a protocol suggested in 2006 [50], which used sequences of coherent states as the public key, as well as optical multiports to implement the distributed swap operations. Extending this protocol to more than three participants would be both theoretically and experimentally challenging due to the complexity of the required multiports. Nevertheless, based on this protocol the first experimental demonstration of a quantum USS scheme was given in 2012 [51], albeit with the assumption that the messaging stage takes place immediately after the distribution stage. This assumption was necessary since the protocol did not remove the requirement of long-term quantum memory.

[8]In the context of computing algorithms, linear scaling is often seen as efficient. However, in the context of signatures it is desired that the size of the signature is much smaller than the size of the message, since appending the signature adds a communication overhead. The signature length of most commonly used digital signature schemes (e.g. RSA, DSA and ECDSA) is small (a couple of kilobytes) and constant with respect to the size of the message being signed. The signature length of some of the most practical USS schemes (e.g. Ref. [42] which use a trusted authority) increases logarithmically in the length of the message being signed. Ideally, our USS schemes would have similar scalings.

Progress on this front was made in Ref. [52], where a protocol was proposed in which participants immediately measure the quantum states they receive and then store only the associated classical information. The measurement suggested was an unambiguous state discrimination (USD) measurement, the outcome of which either identifies the state perfectly, or fails and gives no information.

The use of stored classical information is in some ways less efficient for verifying signatures when compared to the original quantum states, but security can still be shown to be exponential in the length of the sequence of coherent states sent by the sender. The scheme could technically be called a secret-key scheme (as opposed to public-key), since each participant holds a different secret classical key used to check the signature received from the sender. However, this is not a disadvantage when compared to other USS schemes, since all USS schemes are also necessarily interactive and require shared secret keys, so do not have the universal verifiability property. A three-party implementation of this scheme was performed in [53], though authentication of the quantum channels used to transmit the key was not actually performed.

Finally, protocol P1 in Ref. [1] simplified these protocols and removed the necessity of the multiport. Instead, the authors realised that the swap tests could be replaced by a classical post-processing step in which participants exchanged a pre-set number of their classical outcomes. The removal of the multiport, and consequently the removal of the technical difficulties associated with keeping it aligned, led to significant efficiency gains of approximately four orders of magnitude when signing a 1-bit message over 1 km to a security level of 10^{-4}.

Classical vs Quantum

Given the existence of classical USS schemes, one may wonder whether quantum USS schemes are necessary, and what advantages, if any, they may offer over classical schemes. To motivate the use of quantum mechanics, we first note that it provides a unique toolbox that is proven to be well suited to cryptography. We have already seen how unconditionally secure quantum one-way functions can be used to create "public-key" quantum USS schemes. Further, we note that all USS schemes (classical and quantum) require participants to share secret keys with information-theoretic security. This cannot be done classically, but can be done via QKD, and so in a sense all USS schemes are at least an application of quantum technologies and the distinction can sometimes become blurred. For example, the classical USS protocol P2, introduced in Ref. [1], was discovered as part of research into quantum USS

schemes. There, the authors assume that participants distribute secret key using QKD. However, unlike more distinctively "quantum" USS schemes, P2 could proceed identically without using quantum mechanics if given a classical secret key as a resource. In P2, quantum mechanics is only used to generate a secret key, and it therefore seems more appropriate to consider it as a classical USS scheme, otherwise *all* USS schemes would be quantum USS schemes.

Historically it was also believed that quantum USS schemes might be able to achieve the same functionality as classical USS schemes while making fewer assumptions. In any cryptographic protocol, assumptions are crucial to the practical viability and security of the scheme. Refs. [37, 39] assume an authenticated broadcast channel, secret authenticated classical channels, and sufficient secret shared key (required pairwise between all participants) to perform the untraceable sending protocol. These resources are expensive: the secret channels each require shared secret keys of the same length as that of the messages being transmitted [7]; while the authenticated broadcast channel incurs significant communication overhead and is only achievable if fewer than $1/3$ of the participants are dishonest [54]. Refs. [40–42] assume secret channels and a trusted authority, whose role is to distribute the signing and verification keys to each participant. The inclusion of a trusted authority is a large trust assumption, and makes the protocol vulnerable to targeted attacks against the trusted authority, or even to dishonesty or incompetence on the part of the trusted authority. For this reason, and since we want our schemes to be highly secure, this thesis focuses on schemes that do not require a trusted authority.

The quantum USS protocols in Refs. [1, 50, 52] do not assume either a broadcast channel, anonymous channels or the existence of a trusted authority, and are capable of maintaining security as long as the majority of participants are honest. Further, some quantum USS schemes are able to partially remove the need for secret classical channels by employing untrusted quantum channels instead. This led to the belief that quantum USS schemes could achieve the same security guarantees as classical USS schemes while using fewer resources. However, protocol P2 [1] showed that classical USS protocols exist using the same resources as all known quantum USS schemes. In Chapter 8 of this thesis we further disprove this belief by describing an extremely efficient classical USS scheme using significantly fewer resources than even P2 – resources that we prove in Chapter 5 are minimal and necessary for *any* USS scheme that does not use a trusted authority.

2.4 Quantum USS schemes in detail

We use this section to outline in detail a variant of the scheme P1 proposed in Ref. [1]. Throughout this thesis we will refer to this variant as Protocol 1. The scheme forms the starting point for the work in this thesis and is helpful in motivating the chapters to come.

As in all quantum USS schemes proposed in the literature up to 2014, the following protocol describes and analyses the simplest case in which there are only three parties – a sender, Alice, and two recipients, Bob and Charlie – who aim to sign a single 1-bit message. However, unlike previous protocols, the authors move away from coherent states, instead opting to phrase the protocol in terms of the single-photon BB84 states: $|0\rangle$, $|1\rangle$, $|+\rangle := \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and $|-\rangle := \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. The reason for using these states is that they are finite dimensional, and thus they are easier to work with from a theoretical viewpoint. Also, quantum cryptography using single-photons has been widely studied in other contexts, as we shall see in Chapter 3. The switch allowed the authors to leverage existing techniques from relativistic quantum bit commitment [55] to provide a full proof of security against forging – something that in previous protocols [50, 52] had remained elusive. The disadvantage of using single-photon states is that they are experimentally challenging to create, and their use greatly reduces efficiency in a practical setting.

2.4.1 Protocol 1

As usual, the protocol contains two stages: a distribution stage and a messaging stage. The distribution stage sets up the protocol and allows for signed messages to be sent securely at any future date. The authors assume the resources available in the standard resource model, as well as authenticated quantum channels. More concretely, the resources assumed for the security analysis of this protocol are:

- Authenticated classical channels between all participants: this means that all participants (pairwise) must be connected by classical channels and share a short secret key for use in an unconditionally secure message authentication code (MAC), such as Wegman-Carter authentication [56]. The secret key required is normally logarithmic in the length of the message to be authenticated.

- Authenticated Alice-Bob and Alice-Charlie quantum channels: this means that these participants must be connected by quantum channels and share a secret classical key that is at least double the length of the message to be signed.

The secret key is used to encrypt and authenticate the quantum channel [29], though in practice this can be challenging.

- A secret Bob-Charlie classical channel: this means that Bob and Charlie must share a secret key for use in an unconditionally secure encryption protocol such as the one-time-pad. The length of the required secret key is the same as the bit length of the information being transmitted [7].

The messaging stage is much simpler and requires only authenticated classical channels between the senders and receivers of the message. Validation of the message is non-interactive.

The distribution stage

1. For each future 1-bit message $m = 0$ and 1, Alice randomly chooses a secret classical string of symbols to be her private key. $\text{PrivKey}_m = (b_1^m, \ldots, b_L^m)$, where each $b_l^m \in \{0, 1, +, -\}$. The length L will depend on the desired security level of the protocol.

2. For each private key, Alice generates two copies of the sequence of states, $\text{QuantKey}_m = \bigotimes_{l=1}^{L} |b_l^m\rangle \langle b_l^m|$.

3. For each m Alice sends one copy of QuantKey_m to Bob and one to Charlie.

4. For each incoming state, Bob and Charlie randomly and independently choose a basis: either the Z basis $\{|0\rangle, |1\rangle\}$; or the X basis $\{|+\rangle, |-\rangle\}$. They measure the incoming state in that basis. The effect of the measurement is to discover what the state *is not*, e.g. an outcome of $|0\rangle$ definitively rules out $|1\rangle$, but does not rule out either of the other three states as possibilities. This type of measurement is called an *unambiguous state elimination* (USE) measurement. Knowing what the states sent by Alice *are not* allows the recipients to check her signature without being able to recreate it.

5. For each element of each quantum key, Bob and Charlie store the classical description of the ruled out state. They therefore store the triplets $\{m, l, d\}$, where $m \in \{0, 1\}$, $l \in \{1, \ldots, L\}$ and $d \in \{0, 1, +, -\}$, with d recording the *excluded* state. We call the L triplets held by Bob B^m, and the L triplets held by Charlie C^m.

6. Once all states have been received and all triplets recorded, Bob and Charlie each randomly split their keys into two equal parts to obtain the sets B_1^m, B_2^m,

C_1^m and C_2^m, each containing $L/2$ triplets. Using a secret classical channel, they each forward the set indexed 2 to the other participant so that Bob holds B_1^m and C_2^m, while Charlie holds C_1^m and B_2^m. These sets form their private keys and will be used to check future message declarations.

The messaging stage

1. To send the signed 1-bit message m, Alice sends $(m, \text{PrivKey}_m)$ to the desired recipient.

2. Suppose Alice sends the message to Bob. Bob checks whether $(m, \text{PrivKey}_m)$ matches his stored private key. In particular, for each position (indexed by l) in B_1^m and C_2^m, he checks that the excluded state, d, does not equal the corresponding declared element in PrivKey_m. A mismatch occurs in position l^* if $(m, l^*, d^*) \in B_1^m$ and Alice declares $b_{l^*}^m = d^*$ in PrivKey_m. This corresponds to Bob having eliminated the state that Alice claims to have sent. Bob checks for mismatches in B_1^m and C_2^m separately.

3. If the number of mismatches is below $s_a L/2$ for both B_1^m and C_2^m, where s_a is an authentication threshold[9], then Bob accepts the message. If the number of mismatches between PrivKey_m and either B_1^m or C_2^m are more than this threshold, he rejects the message.

4. To forward the message to Charlie, Bob forwards to Charlie the pair $(m, \text{PrivKey}_m)$ that he received from Alice.

5. Charlie tests for mismatches similarly to Bob but uses the sets C_1^m and B_2^m instead. To protect against repudiation and to ensure transferability, Charlie uses a different threshold parameter, s_v, such that $s_v > s_a$. Charlie accepts the message only if both of his sets have fewer than $s_v L/2$ mismatches.

One can see that, if all participants are honest and the channels/detectors are of sufficient quality, then the protocol works correctly – a message sent by Alice would be accepted as valid by both Bob and Charlie.

[9]Of course, in the ideal setting we could choose $s_a = 0$. We choose $s_a > 0$ to allow for channel/detector noise.

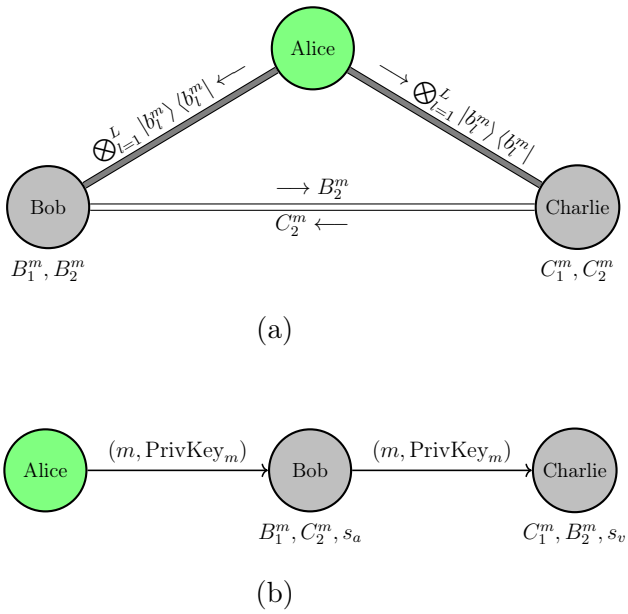


Figure 2.3: The figure shows the distribution stage and messaging stage of Protocol 1. Figure (a) shows a schematic representation of the distribution stage of Protocol 1. The grey Alice-Bob and Alice-Charlie links represent the authenticated quantum channels through which Alice sends QuantKey_m. Upon receiving each state, Bob and Charlie perform an USE measurement (Step 3) to obtain a classical outcome. They use the secret classical channel (represented by the double lines) to exchange half of their measurement outcomes (Step 5). Figure (b) shows a schematic representation of the messaging stage of Protocol 1 in which Alice chooses to send a message to Bob and the message is subsequently transferred to Charlie. The thin black lines represent authenticated classical channels. Bob checks the signature PrivKey_m against B_1^m and C_2^m using a tolerable error threshold of s_a. Charlie checks the forwarded signature against C_1^m and B_2^m using a tolerable error threshold of s_v.

2.4.2 Security

In this subsection we outline the protocol security analysis. To do this, we consider how a dishonest party, Eve, could seek to cheat. The protocol involves only three participants and as such it is assumed that at most one of the participants is dishonest, since two colluding participants could trivially cheat. Since we do not know who is dishonest, the security proof is separated into two parts and considers each possible scenario.

First, we prove that the scheme is secure against forging attempts by imagining that Eve is Bob. Of course, a potential forger could be an external party, but since Bob has access to inside information it is easier for him to cheat than for any external party. Therefore security against forging in the case when Eve is Bob implies security against forging when Eve is an external party. The case when Eve is Charlie is identical.

Second, we imagine that Eve is Alice and prove that she cannot make Bob and Charlie disagree as to the validity of a message she sends. For the three-party scenario we consider this implies both message transferability and non-repudiation.

For now, we work with the informal notions of security provided in Section 2.3.2. We say that a protocol is secure against a particular threat if the probability of an adversary being successful in that dishonest behaviour decays exponentially in the length of the signature, L. The security level of the protocol is defined to be the minimum ϵ such that

$$\epsilon \geq \max\{\mathbb{P}(\text{Forge}), \mathbb{P}(\text{Repudiate}), \mathbb{P}(\text{Non Trans})\}, \tag{2.3}$$

where $\mathbb{P}$(Forge) is the probability of Bob successfully forging a message, $\mathbb{P}$(Repudiate) is the probability of Alice successfully repudiating a message, and $\mathbb{P}$(Non Trans) is the probability of Alice sending a message that is not transferable.

Unforgeability

If Bob wants to forge, he wants to forward a valid message-signature pair to Charlie pretending that it originated with Alice. To do this, he needs to declare a classical string that has fewer than $s_v L/2$ mismatches with C_1^m.[10] Remember, C_1^m is an indexed list of the eliminations arising from Charlie's measurements on the quantum states sent to him by Alice in step 3 of the distribution stage. Helping Bob in his desire to forge is the fact that he holds a valid copy of each of the states measured by Charlie, since he received the same states directly from Alice in step 3 of the distribution stage. A dishonest Bob need not make the measurements specified by the protocol, and he can instead use these states in any way to help create a classical string that does not contradict Charlie's measurement outcomes contained in C_1^m. The difficulty faced by Bob is that he does not know what measurements Charlie performed, and as far as he is aware Charlie could have ruled out any of the three states not sent by Alice. Therefore, to make declarations that will certainly not cause a mismatch with C_1^m, Bob must know the exact identity of each state sent by Alice. Since the states in the ensemble $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$ are non-orthogonal, it is impossible for Bob to distinguish them with certainty. This is essentially the reason why Bob cannot forge a message.

For the purposes of rigorously proving unforgeability, it is useful to categorise the possible attack strategies available to Bob in a similar manner to QKD[11]. The authors define the following types of attack:

[10] Of course, he also needs it to make fewer than $s_v L/2$ mismatches with B_2^m, but since he created B_2^m, we assume this is easy.

[11] Since the stages and aims of a signature protocol are different to those in QKD, the definitions are not identical.

- Individual attacks – Bob measures each state received from Alice separately. The choice of measurement is independent of all previous measurement outcomes.

- Collective attacks – Bob measures each state received from Alice separately. The choice of measurement can depend on previous measurement outcomes.

- Coherent attacks – Bob can make any measurement allowed by quantum mechanics on the global system of all states received from Alice and any auxiliary systems he chooses to create.

The authors formalise the intuitive security arguments above and provide a full proof of security against coherent forging attempts. The authors begin by assuming that Bob is restricted to performing individual attacks. Using cost matrix analysis techniques described in Ref. [57] it is shown that the optimal individual attack for Bob is to use the honest measurement to attempt to determine the state received from Alice. Using this strategy, each element of Bob's forged signature will have an error probability of 1/8 when checked by Charlie. A simple application of the Hoeffding inequality [58] then shows that [59]

$$\mathrm{P}(\text{Forge}) \leq \exp\left[-\left(\frac{1}{8} - s_v\right)^2 L\right]. \tag{2.4}$$

Therefore, as long as $s_v < 1/8$ it is highly unlikely that Bob is able to successfully forge a message. Since the states sent by Alice are independently chosen, convexity arguments show that, for this protocol, the optimal collective attack is actually the same as the optimal individual attack. Lastly, the authors borrow the teleportation strategy technique from relativistic bit commitment [55] to show that even coherent attacks can do no better than individual attacks, meaning the bound derived for individual attacks also holds for arbitrary coherent attacks.

Aside

In general, security is very difficult to analyse when Eve's interaction is not separable[12] (i.e. individual or collective) and quantum USS schemes are no exception. One may wonder why this is the case – since Alice selects and sends each bit independently, is it not obvious that the optimal attack will be an individual one? In

[12]Indeed, for QKD it wasn't until 1996 that a security proof against general coherent attacks was provided [60].

fact, this intuition is wrong, and coherent attacks can be much more powerful than individual ones.

To help understand why, let us follow Ref. [1] and consider more carefully Bob's aims: Bob is trying to declare a signature that makes fewer than a threshold number of errors with C_1^m. In that case, Bob's optimal strategy may not be to try to guess the exact identity of the states sent by Alice. For example, in a protocol where Bob knows Alice sent one of the set $\{001, 010, 100\}$, if Bob was trying to guess what Alice sent while making at most 1 error, then his best strategy is to guess 000, even though this has a zero probability of being exactly what Alice sent. Similarly, making individual measurements on each state received from Alice may lead to the highest probability of guessing PrivKey_m exactly, but not the highest probability of guessing PrivKey_m up to a certain threshold number of errors.

As a concrete example of when this may occur, consider again Protocol 1, but modified so that the states Alice sends are selected from the non-orthogonal ensemble $\{|0\rangle, |+\rangle\}$. Suppose further that $s_v = 1/2$ so that Bob is trying to make mismatches at a rate smaller than 50%. If Bob performs an individual attack, then his optimal strategy is to apply a minimum error measurement independently to each qubit, leading to a mismatch probability of 0.32 per qubit. This can also be shown to be Bob's optimal strategy if his aim is to guess the states Alice sent exactly. Clearly, there is a small but non-zero probability that Bob causes more than 50% mismatches using this strategy. On the other hand, Bob could apply the coherent strategy, introduced in [61], in which he groups each pair of states sent by Alice and measures using the entangled basis

$$\begin{aligned} |\phi_{++}\rangle &= \frac{1}{\sqrt{2}}\left(|01\rangle + |10\rangle\right) \\ |\phi_{+0}\rangle &= \frac{1}{\sqrt{2}}\left(|0-\rangle + |1+\rangle\right) \\ |\phi_{0+}\rangle &= \frac{1}{\sqrt{2}}\left(|+1\rangle + |-0\rangle\right) \\ |\phi_{00}\rangle &= \frac{1}{\sqrt{2}}\left(|+-\rangle + |-+\rangle\right). \end{aligned} \tag{2.5}$$

The outcome determines Bob's guess, e.g. an outcome of $|\phi_{00}\rangle$ implies Bob guesses $|0\rangle|0\rangle$ as the states sent by Alice. Using this strategy, the probability that Bob gets *both* states wrong is zero for any pair of states sent by Alice. Therefore, Bob will certainly not make more than 50% errors overall, and so this coherent strategy allows Bob to forge with probability 1. This counter intuitive argument displays the power of coherent strategies and highlights the difficulties faced in quantum USS

schemes when considering coherent attacks.

In Chapter 6 we prove a general theorem relating an adversary's smooth min-entropy on PrivKey_m to the minimum number of mistakes she is likely to make when guessing PrivKey_m. The theorem is very useful when considering coherent forging strategies in quantum USS schemes.

Transferability

For proving that messages are transferable, Eve is assumed to be Alice. The aim of a dishonest Alice in this case is to produce a message that will be accepted at the level s_a and rejected at the level s_v. Proving security against this type of attack is often simpler than proving unforgeability, as the argument is essentially classical.

Whatever states Alice sends to Bob, Bob makes his measurements and stores a list of classical measurement outcomes. This classical list will have an error rate e_B with the signature that Alice later sends. Similarly, the outcomes stored by Charlie will have an error rate of e_C with Alice's future signature. The error rates e_B and e_C are unknown, and are totally within the control of Alice, but they are fixed. The exchange process performed in step 6 of the distribution stage effectively means that each recipient will test Alice's signature against one set with mismatch rate e_B, and one set with mismatch rate e_C. More concretely, the symmetrisation involves both recipients selecting $L/2$ triplets from their outcomes without replacement, meaning the number of mismatches selected follows a hypergeometric distribution. The message is likely to be accepted by a recipient only if both e_B and e_C are below the testing threshold (s_a or s_v). Since $s_v > s_a$, it is highly unlikely for the message to be accepted at level s_a by the first recipient and subsequently rejected at level s_v by the second (since passing the first test likely means $e_B, e_C < s_a < s_v$).

Existing results on hypergeometric distribution tail bounds [62] formalise this intuition and show that Alice's optimal strategy is to choose $e_B = e_C = \frac{1}{2}(s_v - s_a)$, in which case

$$\mathrm{P}(\text{Non Trans}) \leq 2\exp\left[-\frac{1}{4}(s_v - s_a)^2 L\right]. \tag{2.6}$$

Non-repudiation

If a dishonest Alice wants to repudiate a message, she wants to deny having sent a message that she actually did send. The notion of repudiation is closely related to transferability, but the exact details rely on the chosen form of dispute resolution. Dispute resolution, formally defined in Chapter 4, can be thought of as a last resort

for participants who do not agree. It should not happen in normal runs of the protocol, and is akin to taking someone to court over breach of contract.

Suppose Charlie receives a message from Bob which is claimed to have originated from Alice. If Alice denies (repudiates) having sent the message, how does one decide who is telling the truth? In such a scenario, if neither party backs down the dispute resolution process is triggered to decide who is telling the truth. Dispute resolution should always decide in favour of honest participants. The dispute resolution procedure used throughout this thesis is "majority vote", in which all participants get together and vote whether the message is valid or not, with the final decision going to the majority. In our scenario, if Alice actually did send the message, then transferability ensures both Bob and Charlie would find the message to be valid and so the majority vote would be "valid". If Alice did not send the message, then security against forging would ensure that Charlie would find the message invalid, so the majority vote would be "invalid". Thus the protocol is secure against repudiation attempts and

$$\mathbb{P}(\text{Repudiation}) = \mathbb{P}(\text{Non Trans}). \tag{2.7}$$

2.4.3 Experimental implementation

The protocol above was implemented experimentally in Ref. [59] using the coherent state ensemble $\{|\alpha\rangle, |i\alpha\rangle, |-\alpha\rangle, |-i\alpha\rangle\}$ rather than the single-photon states used for the theoretical analysis. In switching to coherent states, the choice of the mean photon number α becomes important, since larger values reduce errors and loss, but also allow for easier cheating as the states in the ensemble become more distinguishable. The experiment found $\alpha = 0.4$ to be optimal.

The protocol was performed over distances ranging up to 2 km, and required a signature length of $L \approx 10^{10}$ to sign one bit over 1 km to a security level of $\epsilon = 10^{-4}$. Although this signature length is long, the switch to coherent states facilitates the use of readily available high clock-rate sources, meaning messages can be signed at the rate of approximately one bit every 20 seconds. This is a vast improvement over the previous multiport schemes, which required $L = O(10^{13})$ to sign a 1-bit message over a distance of just a few metres, translating to a signing rate of roughly one bit per eight years!

The implementation in Ref. [59] did not exactly recreate the theoretical model analysed in Ref. [1]. Instead, a new security analysis was required and unfortunately it did not prove information-theoretic security against all types of attack. Specifically, the implementation was not proven secure against coherent forging attempts.

The security analysis contained in Ref. [1] depended heavily on the fact that Alice sends single-photon states and it was not clear how one could modify the analysis to cover the case of Alice sending coherent states. A further drawback of the implementation was that due to the difficulty of implementing authenticated quantum channels, they were not used in this experiment. Instead, the experiment made the simplifying assumption that Eve does not in any way eavesdrop or tamper with the states sent over the quantum channels to Bob and Charlie. Effectively, Eve (who for the purposes of forging is either Bob or Charlie) is restricted to strategies where she interacts only with the legitimate states received from Alice in step 3 of the distribution stage. Of course, to claim that the implementation is fully unconditionally secure these issues would need to be resolved.

2.4.4 Summary

The protocol presented in this section represents the culmination of research into quantum USS schemes as of 2014. The protocol does not require quantum memory, thereby removing issue (II) of the original Gottesman-Chuang scheme. Protocol 1 also partly alleviates drawback (I), as the theoretical model required only a source of single-photon states, something that is achievable with current technology. In practice however, requiring a single-photon source would make the protocol extremely inefficient and so the "public key" QuantSig_m was instead implemented using a sequence of coherent states. Unfortunately, this compromise left a gap between theory and experiment, and there was no longer a full security proof. Protocol 1 does not address issues (III) - (VI).

Protocol 1 is also highly inefficient, and to reduce Eqs. (2.4), (2.6) and (2.7) to be $< 10^{-4}$ requires an extremely large signature length, L, and very large secret shared key requirements between each pair of participants. More concretely, to sign each and every 1-bit message over 1km recipients needed to share (pairwise) approximately 10^{10} secret bits. In addition, the signature attached to each $1-$bit message would be 8 GB in size. This is wholly impractical, since to transmit a signed 1 MB message between just three participants over 1km would require attaching a 1 Petabyte signature!

2.5 Thesis goals

In light of the state of USS schemes and the discussions contained within this chapter, we identify some clear research goals that will form the basis of this thesis.

- *We would like to find a scheme that is both experimentally implementable and has a full proof of unconditional security.* There are two possibilities: either bring the experiment in line with the theory, or bring the theory in line with the experiment. In Chapter 6 we do the latter; we present an unconditionally secure error-tolerant quantum USS scheme that can be implemented using only coherent states, and that does not rely on authenticated quantum channels or additional trust assumptions. The resulting protocol fully addresses issues (I) and (II). Chapter 7 goes one step further, and describes the first measurement-device-independent quantum USS scheme, thereby removing common side-channel attacks arising in real-world implementations.

- *Continue to investigate what is possible with unconditional security in the quantum setting, and where quantum mechanics may provide a benefit over classical schemes.* This question is addressed mainly in Chapter 8, where the resource requirements and trust assumptions required in both classical and quantum USS schemes are compared and considered in detail.

- *Investigate the relationship between USS schemes and other cryptographic protocols such as oblivious transfer, Byzantine agreement and QKD.* This is an interesting open question with many possible avenues of research. In Chapter 3 we outline the relationship of USS schemes to Byzantine agreement and oblivious transfer. In Chapter 6 we consider the relationship between quantum USS schemes and QKD, and find some interesting similarities and differences. Finally, in Chapter 9 we investigate the potential application of imperfect oblivious transfer to USS schemes.

- *Increase the efficiency of USS schemes without making additional resource assumptions.* Specifically, we would like to decrease both the shared secret key requirements and the length of the signature, while remaining within the standard resource model. Chapter 8 describes and analyses an N-party classical USS scheme which is vastly more efficient than all previous USS schemes, both in terms of signature size and secret shared key requirements. In terms of resource requirements, as well as falling within the standard resource model, the secret shared key needed between participants scales similarly to message authentication, meaning it is extremely cheap to implement compared to all other USS schemes. Additionally, the scheme fully resolves drawbacks (I) - (V) and can be considered practical.

Chapter 3

Quantum cryptography

3.1 Introduction

In this chapter we introduce various concepts in quantum and classical cryptography that will be useful for understanding the material presented in later chapters. Many of the results we discuss emerged from the theoretical analysis of quantum key distribution, and are therefore most naturally introduced in that context. As such, we begin this chapter with a detailed look at QKD protocols. Nevertheless, the analytic techniques developed are widely applicable and have been central to the study of many other cryptographic protocols. One such example is the quantum USS scheme presented in Chapter 6 of this thesis, whose security analysis partially relies on two important and deep concepts in quantum information theory – the data processing inequality and entropic uncertainty relations – both of which are discussed here.

We also use this chapter to introduce classical authentication schemes providing unconditional security. Though often overlooked, classical authentication is always required for cryptographic protocols to be secure in the information-theoretic setting. Authenticated communication is necessary to prevent powerful *man-in-the-middle* attacks in which an adversary can intercept, modify or create information while pretending to be a legitimate protocol participant. Authentication schemes, though distinct from USS schemes, are closely related both in terms of their aims and their construction. In Chapter 8 we modify authentication schemes to create a new and highly efficient classical USS scheme using an *almost strongly universal* set of hash functions. By design, almost strongly universal sets are well suited for use in one-time USS schemes.

Lastly, we briefly introduce two concepts – oblivious transfer and Byzantine

agreement – both of which seem closely related to signatures, though the relationship has not been fully explored. Oblivious transfer is one of the most important primitives in modern cryptography, with a variety of applications including secure multiparty computation, oblivious sampling, e-voting and many more. The distribution stage of many USS schemes requires a sender to distribute partial information out to many recipients, with security guarantees reminscent to those of oblivious transfer. The potential application of oblivious transfer to USS schemes is considered. Byzantine agreement, also known as authenticated broadcast, is a problem that often arises in the context of distributed computing and fault tolerance. The aims of Byzantine agreement are similar to those of USS schemes, but there are important differences which we highlight at the end of this chapter.

3.2 Notation

We describe a d-dimensional pure quantum state by a vector, $|\phi\rangle$, in a d-dimensional Hilbert space $\mathcal{H}$. More generally, states can be described by density operators, ρ, which are normalised ($\mathrm{Tr}(\rho) = 1$) positive semi-definite Hermitian operators acting on vectors in $\mathcal{H}$. We denote the space of density matrices as $\mathcal{D}(\mathcal{H})$. In some cases, we will refer to the space of sub-normalised ($\mathrm{Tr}(\rho) \leq 1$) positive semi-definite Hermitian operators acting on vectors in $\mathcal{H}$. We denote this space as $\mathcal{D}_{\leq}(\mathcal{H})$.

A measurement on a quantum system is described in general by a collection of positive semi-definite Hermitian operators $\mathcal{M} = \{M_x\}_{x\in\mathcal{X}}$ which act on a Hilbert space, and which sum to the identity operator on that Hilbert space, i.e. $\sum_x M_x = \mathbb{1}$. A measurement $\mathcal{M}$ is called a POVM, while the individual operators M_x are called the POVM elements. We denote the space of positive semi-definite Hermitian operators acting on vectors in $\mathcal{H}$ as $\mathcal{P}(\mathcal{H})$. Note that $\mathcal{D}(\mathcal{H}) \subset \mathcal{D}_{\leq}(\mathcal{H}) \subset \mathcal{P}(\mathcal{H})$. Often the scenarios we consider involve composite systems that have both a classical and quantum element, and for this it is useful to define the notion of a classical-quantum state.

Definition 3.1 (Classical-quantum states)**.** A state ρ_{XA} is called a classical-quantum state, or cq-state, if it has the form

$$\rho_{XA} = \sum_x p(x) \left|x\right\rangle \left\langle x\right|_X \otimes \rho_A^x, \tag{3.1}$$

where $\{|x\rangle\}_x$ is an orthonormal basis and ρ_A^x is a normalised density matrix for all values of x.

3.3 Quantum key distribution

A basic requirement of many unconditionally secure cryptographic protocols is that two recipients share a secret key – i.e. they share a bit string that is kept secret from everyone else. For encryption, Shannon proved that encrypting a message with information-theoretic security requires a secret key that is at least as long as the message being encrypted [7]. The one-time pad [32] is an example of an encryption scheme providing information-theoretic security, as long as the key is kept secret. As a result of Shannon's theorem, sending messages with perfect secrecy requires a large shared secret key.

Of course, for a scheme to provide information-theoretic security the encryption key must be generated in an information-theoretically secure way. Classically this is an impossible task; one cannot use purely classical means to generate shared secret randomness with unconditional security. On the other hand, the inherent randomness of quantum mechanics is a powerful cryptographic resource which allows for protocols that accomplish previously impossible tasks, such as unconditionally secure key distribution.

3.3.1 The protocol

QKD is a protocol run by two honest participants – Alice and Bob – to generate a random and perfectly secret key known to each participant. For the purposes of the security analysis, and since we are looking for unconditional security, we assume there is an eavesdropper (Eve) who is completely unbounded, except for the physical restraints imposed by quantum mechanics. We assume that Alice's and Bob's labs are secure, but that Eve is free to operate everywhere else; specifically, this means Eve cannot interfere with state preparation or measurement, but can interact with anything transmitted between Alice and Bob. The goal of the eavesdropper is to discover all or some of the secret key sent from Alice to Bob.

Prepare-and-measure BB84

A simple prepare-and-measure BB84 QKD protocol [10] can be described using single-photons, with information encoded into one of two bases: the X basis $\{|+\rangle, |-\rangle\}$, or the Z basis $\{|0\rangle, |1\rangle\}$, where $|\pm\rangle = 1/\sqrt{2}(|0\rangle \pm |1\rangle)$. The protocol proceeds as follows:

1. Alice selects a sequence of uniformly random bits $x_1, x_2, \ldots$. This sequence is Alice's *raw key*.

2. Alice encodes each bit into a quantum state using a basis chosen uniformly at random, i.e. Alice encodes 0 as $|0\rangle$ or $|+\rangle$, and 1 as $|1\rangle$ or $|-\rangle$, depending on her choice of basis.

3. Alice transmits the states over the quantum channel to Bob.

4. Bob measures each incoming state using a basis chosen uniformly at random, either X or Z. The sequence of measurement outcomes forms Bob's raw key.

5. Once all states have been measured, Alice and Bob each publicly announce the bases they used for preparation/measurement. All outcomes arising from states prepared and measured using different bases are discarded. The remaining keys held by Alice and Bob are called the *sifted keys*.

6. Alice and Bob agree to sacrifice a subset of their sifted keys in order to estimate the error rate between them. We call this procedure *parameter estimation*. If the error rate is too high, the protocol is aborted.

7. Alice and Bob perform classical post-processing on their remaining sifted keys to correct errors and enhance privacy.

Note that all classical communication is performed over authenticated (but not secret) channels.

Entanglement-based BB84

From a security analysis perspective, the BB84 protocol above can be equivalently described by a scheme in which maximally entangled states are distributed and subsequently measured by each party [63]. These schemes are often referred to as EPR schemes, or entanglement-based schemes, due to their similarity to the EPR paradox [4].

In entanglement-based schemes, Alice prepares n copies of the two qubit state

$$\left|\Phi^+\right\rangle = \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right) = \frac{1}{\sqrt{2}}\left(|{+}{+}\rangle + |{-}{-}\rangle\right), \tag{3.2}$$

and sends the second qubit in each state to Bob. Once Bob has received all qubits Alice uniformly at random selects a basis for each state, either X or Z, and publicly announces her choice to Bob. They each measure the states using the publicly chosen basis. The results of Alice's measurements form a key, X, and the results of Bob's measurements form a key, $\tilde{X}$. These are the sifted keys, because they are the keys held *before* the classical post-processing has taken place.

The mathematical equivalence between this protocol and the prepare-and-measure version of BB84 follows by noticing that Alice's measurements on her states commute with the transmission of Bob's states over the quantum channel, and can therefore be performed before transmission without altering security. The only remaining difference is that, in the prepare-and-measure version of the protocol, Alice and Bob must post-select only those states for which the correct basis was chosen (i.e. they must perform basis reconciliation). For the theoretical analysis of QKD, it is usual to work with the entanglement version of the protocol, while for experimental implementations, the prepare-and-measure version is much more practical.

3.3.2 Security overview

Security follows from the monogamy of entanglement, together with the parameter estimation procedure used to gauge the level of correlation between X and $\tilde{X}$. It can be shown that $|\Phi^+\rangle$ is the only state for which Alice and Bob are guaranteed to obtain perfectly correlated results when they measure in *either* the X or the Z basis. Therefore, if they are able to ascertain that their results are always perfectly correlated, they can deduce that, even after transmission, they must have shared the state $|\Phi^+\rangle$. By the monogamy of entanglement, since $|\Phi^+\rangle$ is a maximally entangled state it cannot be entangled with any other state. Therefore, Eve cannot have any information on the generated key. Finite-size, noise tolerant versions of this argument can also be shown to hold as long as X and $\tilde{X}$ are sufficiently correlated.

Classical post-processing

Real-world implementations of QKD inevitably involve errors in the preparation, transmission and detection of states. This means Alice's sifted key X will not equal Bob's sifted key $\tilde{X}$. As such, Alice and Bob perform a classical error correction protocol on their sifted keys. The goal of error correction is for Alice to send a minimal quantity of information, C, such that given $\tilde{X}$ and C, Bob can reconstruct X exactly (with high probability) [64]. Following this, Alice and Bob each hold the partially secret key X.

Channel imperfections leak information to the environment (Eve). Additionally, the error correction protocol is performed over public classical channels, meaning the extra information C is also available to Eve. The result is that X is only partially secure, and the participants must perform additional post-processing to enhance security.

Privacy amplification is a form of randomness extraction. The goal is for Alice to send information R to Bob so that both Alice and Bob can use X and R to generate a shorter key Z. This is done in such a way that without knowledge of X (but even with all other eavesdropped knowledge), the adversary has no information on Z except with negligibly small probability [64]. For unconditional security against quantum adversaries, privacy amplification is done via universal hashing (see Section 3.6.2).

Below we introduce various distance metrics and entropy measures which are useful for examining the security and efficiency of error correction and privacy amplification.

3.3.3 The trace distance

In order to be able to say that QKD is unconditionally secure, we first need to define in a rigorous way precisely what "unconditionally secure" means in this context. To do this, we introduce a metric called the *trace distance*. This metric is used extensively in quantum cryptography and quantum information theory.

Definition 3.2 (Trace distance)**.** Let $\rho, \sigma \in \mathcal{D}(\mathcal{H})$ be two normalised density matrices. The trace distance $T(\rho, \sigma)$ is defined as half of the trace norm of the difference of the two states, i.e.

$$T(\rho, \sigma) := \frac{1}{2}||\rho - \sigma||_1 = \frac{1}{2}\mathrm{Tr}\left[\sqrt{(\rho - \sigma)^\dagger(\rho - \sigma)}\right]. \tag{3.3}$$

This definition can also be extended to a metric on sub-normalised density matrices, as in Ref. [65]. The trace distance is the quantum generalisation of the Kolmogorov distance for classical probability distributions. A useful operational interpretation of the trace distance is that it quantifies the maximum probability of distinguishing between two states when using an optimal measurement. Specifically, if one wants to distinguish between two states, ρ and σ, the optimal strategy for doing so would have success probability $\frac{1}{2}(1+T(\rho, \sigma))$ [66]. The trace distance takes values in the range $[0, 1]$, with larger values meaning the states are more distinguishable. The trace distance is used to define the security of QKD.

Definition 3.3 (QKD security [67])**.** Suppose that the output of a QKD protocol is ρ_{ZE}, where Z is the classical key held by Alice and Bob, and $\mathcal{H}_E$ is the Hilbert space containing Eve's potentially correlated information. The QKD protocol is

called ϵ-secure if

$$T(\rho_{ZE}, \rho_U \otimes \rho_E) \leq \epsilon, \tag{3.4}$$

where $\rho_U := \sum_{z \in Z} |z\rangle \langle z|$ is the maximally mixed state on the key space.

The meaning of this definition is that, except with probability ϵ, the protocol outputs a state that is indistinguishable from a uniformly random key that is completely uncorrelated with Eve. In this case, Eve can do no better than to randomly guess Z.

3.3.4 The purified distance

Another useful distance measure in quantum cryptography is the *purified distance*. To define the purified distance, we first need to introduce the concept of *fidelity*.

Definition 3.4 (Fidelity)**.** Let $\rho, \sigma \in \mathcal{D}(\mathcal{H})$ be two normalised density matrices. The fidelity is defined via the trace norm as

$$F(\rho, \sigma) := ||\sqrt{\rho}\sqrt{\sigma}||_1. \tag{3.5}$$

Similarly to trace distance, the fidelity takes values in the range $[0, 1]$, but now larger values mean that the states are less distinguishable. The fidelity itself is actually not a distance metric on the space of normalised states, but is closely related to the trace distance according to the Fuchs-van de Graaf inequalities [68]

$$1 - F(\rho, \sigma) \leq T(\rho, \sigma) \leq \sqrt{1 - F(\rho, \sigma)^2}. \tag{3.6}$$

The upper bound becomes an equality if the states are pure. The fidelity can be used to define other useful distance metrics, one of which is the purified distance.

Definition 3.5 (Purified distance)**.** Let $\rho, \sigma \in \mathcal{D}(\mathcal{H})$ be two normalised density matrices. The purified distance is defined as

$$P(\rho, \sigma) := \sqrt{1 - F(\rho, \sigma)^2} = \min_{|\phi\rangle, |\psi\rangle} T(|\phi\rangle, |\psi\rangle), \tag{3.7}$$

where the minimisation in the second equality is taken over all purifications of ρ and σ.

Note that the second equality in the above definition uses Uhlmann's theorem to express the fidelity as the maximum overlap of the purifications, and then the Fuchs-van de Graaf inequality to replace the fidelity by the trace. The purified distance

can be extended to a metric on the space of sub-normalised density matrices using the concept of *generalised fidelity* [65]. This extension has established itself as an extremely useful tool for proving important results in quantum information theory, such as the duality of min- and max-entropies, and the related uncertainty relations that we will see below. As long as one of the density matrices is normalised, the form of the generalised purified distance stays the same as above.

3.4 Entropy

In this section we introduce various entropic measures that have proved particularly useful in the study of many cryptographic protocols. Entropy is important because it quantifies uncertainty, but there is no "one size fits all" measure. Rather, there is a whole family of related entropic quantities, each of which is best suited to describing different resources. The precise meaning of a particular entropic measure is often unclear from the definition alone, and it is therefore always useful to provide an operational interpretation of the measure defined.

3.4.1 The von Neumann entropy

One of the most well-known measures of entropy in a quantum system is the von Neumann entropy.

Definition 3.6 (Von Neumann entropy)**.** Let $\rho_A \in \mathcal{D}(\mathcal{H}_A)$ be a density matrix. The von Neumann entropy of ρ_A is

$$H(A)_\rho := -\mathrm{Tr}[\rho \log(\rho)]. \tag{3.8}$$

Entropy defined in this way is equal to the Shannon entropy of the spectral decomposition of the state. Specifically, if the spectral decomposition of ρ is

$$\rho = \sum_x p(x) \left|x\right\rangle \left\langle x\right|, \tag{3.9}$$

then the von Neumann entropy of ρ is equal to the Shannon entropy of a random variable X distributed according to $p(x)$. Therefore, in analogy to the Shannon entropy, one can think of the von Neumann entropy as the expected information gain upon receiving and measuring a state.

An alternative characterisation is provided by Schumacher's noiseless coding theorem [69, 70], which is directly analogous to Shannon's noiseless coding theorem for

classical information. Suppose Alice chooses n states independently from the ensemble $\mathcal{E}$, and uses a quantum channel to transmit them to Bob. Suppose also that

$$\mathcal{E} = \Big\{ \{p(y_1), |y_1\rangle \langle y_1|\}, \ldots, \{p(y_k), |y_k\rangle \langle y_k|\} \Big\}, \tag{3.10}$$

and that the states $|y_i\rangle$ are not necessarily orthogonal. Note that, since Alice is choosing many states independently from the same ensemble, we are in the realm of asymptotic IID information theory. From Bob's perspective, he receives $\sigma^{\otimes n} = \sigma \otimes \sigma \otimes \cdots \otimes \sigma$, where

$$\sigma = \sum_{i=1}^{k} p(y_i) \, |y_i\rangle \langle y_i| \, . \tag{3.11}$$

Schumacher's noiseless coding theorem states that, in the limit as $n \to \infty$, if Alice wants Bob to be able to decode all states perfectly, she must use the quantum channel to transmit *at least* $nH(\sigma)$ qubits. In other words, the ensemble contains $H(\sigma)$ incompressible qubits of information, since to transmit a single state in the ensemble (so that it can be decoded without error) Alice must asymptotically send an average of $H(\sigma)$ qubits. In this way, just as the Shannon entropy gives rise to the notion of an incompressible bit of information, the von Neumann entropy gives rise to the notion of an incompressible qubit of quantum information.

3.4.2 The conditional quantum entropy

In analogy with the conditional Shannon entropy, the conditional quantum entropy is defined as the difference between the entropy of a joint state and the entropy of its reduced state.

Definition 3.7 (Conditional quantum entropy)**.** Let $\rho_{AB} \in \mathcal{D}(\mathcal{H}_A \otimes \mathcal{H}_B)$ and let $\rho_B \in \mathcal{D}(\mathcal{H}_B)$ be the reduced density matrix of ρ_{AB}. The conditional quantum entropy is defined as

$$H(A|B)_\rho := H(AB)_\rho - H(B)_\rho. \tag{3.12}$$

One must be careful when interpreting the conditional quantum entropy. The definition suggests that it is the additional uncertainty contained in the joint system over and above that contained in the reduced system, or the uncertainty in system A given access to system B. However, a striking departure of quantum information theory from classical information theory is that the conditional quantum information can be negative, meaning this interpretation does not make sense in the quantum setting[1]. Nevertheless, for cq-states the conditional quantum information is always

[1]Instead, the general operational interpretation of conditional quantum information is given by

positive and the naive interpretation makes sense. Below we present three examples of operational interpretations of the conditional quantum information for cq-states in the asymptotic IID setting.

Private randomness extraction

Randomness extraction is closely related to the task of privacy amplification used in QKD. Consider the asymptotic setting in which two honest parties, Alice and Bob, share many IID copies of a classical random variable X, and an adversary holds quantum side information E on each copy. Equivalently, suppose Alice, Bob and Eve share many copies of the state

$$\rho_{XE} = \sum_x p(x) \left|x\right\rangle \left\langle x\right|_X \otimes \rho_E^x, \tag{3.13}$$

where both Alice and Bob have access to the X system, and Eve has access to the E system. The aim is for Alice to send a public message R to Bob such that, using X and R, Alice and Bob can each compute a shorter key Z. The generated key must contain uniform random bits that are uncorrelated with E. This problem was considered in Ref. [72], and the maximum rate at which uniform random bits can be extracted was found to be approximately $H(X|E)_\rho$ [73]. Intuitively, $H(X|E)_\rho$ is therefore the amount of randomness in X that is independent to system E.

The classical-quantum Slepian Wolf problem

The classical-quantum Slepian Wolf (CQSW) problem was considered by Devetak and Winter [74], and concerns classical data compression when the decoder has access to quantum side information. Suppose Alice and Bob share n copies of a cq-system YB. Alice possesses full knowledge of the Y systems, so knows $y_1, \ldots, y_n$, but does not have access to the quantum B systems, which are locally described by $\rho_{y_1} \otimes \cdots \otimes \rho_{y_n}$. Bob has access to the quantum B systems, but not the classical Y systems. Alice aims to send Bob information at a minimal rate allowing him to reconstruct the Y values.

In the case of asymptotically large n, Alice must send Bob information at a rate of at least $H(Y|B)_\rho$ bits per copy for Bob to be able to perfectly reconstruct the classical data $y_1, \ldots, y_n$ using only B and the information received from Alice. Intuitively, Bob's uncertainty on each Y system is therefore quantified as $H(Y|B)_\rho$

the task of state merging [71]

since, for Bob to know each Y with certainty, Alice must reduce his uncertainty on each Y to zero, meaning she must send him at least $H(Y|B)_\rho$ bits of data per copy.

This result generalises the classical Slepian Wolf problem [75], which is relevant to the classical protocol of error correction in QKD. Suppose the sifted keys held by Alice and Bob are generated from n independent measurements on a state of the form $\rho_{AB}^{\otimes n}$. This assumption is commonly made in the analysis of (entanglement-based) QKD protocols, and amounts to restricting the adversary to performing only collective attack strategies. In QKD, a collective attack is one in which Eve interacts identically and independently with each qubit sent over the quantum channel, so that the state shared by Alice and Bob after transmission is in the product form shown[2]. In this case, each of Alice's and Bob's n measurement outcomes can be described by the random variables Y and $\tilde{Y}$ respectively. Alice's sifted key X is the concatenation of the n independent realisations of the random variable Y, and Bob's sifted key $\tilde{X}$ is the concatenation of the n independent realisations of the random variable $\tilde{Y}$. In the error correction phase, Alice sends Bob information to allow him to deduce X using $\tilde{X}$. The CQSW theorem means that Alice must send Bob at least $nH(X|\tilde{X})$ bits of information for error correction to be successful.

One-way secret key distillation

Consider the asymptotic IID setting in which three parties, Alice, Bob and Eve, share many copies of the cqq-state

$$\rho_{XBE} = \sum_x p(x) \left|x\right\rangle \left\langle x\right|_X \otimes \rho_{BE}^x, \tag{3.14}$$

where Alice holds system X, Bob holds system B and Eve holds system E. Again, this scenario applies to QKD when the eavesdropper is restricted to collective attack strategies. The Devetak-Winter bound [76] states that the rate at which Alice and Bob can distill a uniformly random key using one-way communication, such that the key is secret and independent to Eve, is at least

$$H(X|E)_\rho - H(X|B)_\rho. \tag{3.15}$$

This is analogous to the Csiszár-K$\acute{}$orner bound for the secrecy capacity of the memoryless classical wiretap channel [77]. Intuitively, one can think of Alice extracting randomness independent to Eve at a rate of $H(X|E)_\rho$, as per the randomness ex-

[2]To deal with completely general attacks, also referred to as *coherent attacks*, one needs the machinery of one-shot information theory which is introduced in the following sections.

traction subsection, but having to reduce this rate by $H(X|B)_\rho$ to ensure Bob has zero uncertainty on the generated key, as per the CQSW theorem.

3.4.3 One-shot quantum information theory

As can be seen from the sections above, the von Neumann entropy and the conditional quantum entropy are useful for studying IID events in the asymptotic limit. This is often a good approximation for communication protocols in which the sender transmits many messages taken from the same ensemble, and when the channel is approximately memoryless. For cryptography on the other hand, assuming that the channel is memoryless is equivalent to assuming the adversary acts independently on each state, which amounts to imposing a restriction on the adversary's abilities. Further, asymptotically large sample sizes are unrealisable in practice, and any asymptotic analysis may gloss over finite-size effects that the adversary can exploit.

For practical cryptography in the unconditionally secure setting, channels are used a finite number of times and cannot be assumed to be memoryless, since the adversary has memory and can introduce arbitrary correlations between successive states. Protocols in this setting are studied using the tools of *one-shot information theory*, and for this the notions of min- and max-entropy have proved extremely useful. The term "one-shot" is used to distinguish this setting from the asymptotic IID setting. If n states are input into a channel with memory, one cannot treat the inputs as separate events, and one must instead consider the whole process as a *single* input into a larger channel.

Min-entropy

The min-entropy is the smallest Renyi entropy [78] and provides a lower bound on the von Neumann entropy. The conditional min-entropy was first considered in the quantum setting by Renner and Konig [67, 72] and has since been defined as follows.

Definition 3.8 (Conditional min-entropy [79])**.** Let $\rho_{AB} \in \mathcal{D}(\mathcal{H}_{AB})$. The min-entropy of A conditioned on B of the state ρ_{AB} is

$$\begin{aligned} H_{\min}(A|B)_\rho &:= \max_\sigma \sup\{\lambda \in \mathbb{R} : \rho_{AB} \leq 2^{-\lambda} \mathbb{1}_A \otimes \sigma_B\} \\ &:= \max_\sigma H_{\min}(\rho_{AB}|\sigma_B), \end{aligned} \tag{3.16}$$

where the maximisation is taken over all states $\sigma \in \mathcal{D}_\leq(\mathcal{H}_B)$. If the state ρ_{AB} is obvious from the context, the subscript ρ on the min-entropy may be dropped.

For finite dimensional Hilbert spaces, the min-entropy takes values in a closed compact set, meaning the supremum value will always be attained and so can be replaced by a maximisation. Although the conditional min-entropy seems difficult to compute due to the maximisations involved in the definition, it is possible to express the min-entropy as a semi-definite program (SDP), and so find its numerical value efficiently.

Max-entropy

The max-entropy was originally defined by Renner in terms of the 0-order Renyi entropy [67]. However, since then, the definition has been refined to one which is more convenient by virtue of its direct duality with the min-entropy [80].

Definition 3.9 (Conditional max-entropy)**.** Let $\rho_{AB} \in \mathcal{D}(\mathcal{H}_A \otimes \mathcal{H}_B)$ be a bipartite density operator, and let ρ_{ABC} be any purification. The max-entropy of A conditioned on B is defined by

$$H_{\max}(A|B)_\rho := -H_{\min}(A|C)_\rho. \tag{3.17}$$

Note that this definition is consistent because all purifications of ρ_{AB} onto system C are related by a unitary transformation on C. Since $H_{\min}(A|C)_\rho$ is invariant under local unitaries on C, the max-entropy is independent of the choice of purification.

Smooth min- and max-entropy

A useful feature of the von Neumann entropy is that it is continuous – if two states are close according to the trace distance, then their entropies are also close. This is a consequence of the Fannes–Audenaert inequality [81] and it has many useful applications, such as proving converse theorems for quantum channel capacities [69]. The min- and max-entropies defined above have the undesirable property that small changes in the state can cause large jumps in the entropy of the system. Often, we are interested in scenarios and protocols that are only approximately correct, e.g. with probability $1 - \epsilon$. In these cases, we may not be interested in the entropy of the exact state, but rather, the entropy of states close to the exact state. As the name suggests, the ϵ-smooth min/max-entropy "smooths" out the large variations in the min/max-entropy by considering (respectively) the maximum/minimum entropy over a ball of states ϵ-close to the original.

Definition 3.10 (Smooth min/max-entropy [79])**.** The ϵ-smooth min/max-entropy

of A conditioned on B of the state ρ_{AB} is defined as

$$\begin{aligned} H_{\min}^{\epsilon}(A|B)_{\rho} &:= \max_{\overline{\rho}_{AB}} H_{\min}(A|B)_{\overline{\rho}} \\ H_{\max}^{\epsilon}(A|B)_{\rho} &:= \min_{\overline{\rho}_{AB}} H_{\max}(A|B)_{\overline{\rho}} \end{aligned} \tag{3.18}$$

where the maximisation/minimisation is performed over all operators $\overline{\rho}_{AB} \in \mathcal{D}_{\leq}(\mathcal{H}_{\mathcal{A}} \otimes \mathcal{H}_B)$ that are ϵ-close to ρ_{AB} in terms of the purified distance, i.e. $P(\overline{\rho}_{AB}, \rho_{AB}) \leq \epsilon$.

We will see that these smoothed quantities have useful operational interpretations in scenarios when only approximate correctness/security is required.

Min-entropy as a guessing probability

Expressing the min-entropy as an SDP provides a useful operational interpretation on cq-states in terms of the optimal probability of guessing the value of the classical system, given access only to the correlated quantum system [80]. Suppose Alice and Bob share a *single* cq-state ρ_{XB}, with Alice holding the X system and Bob the B system[3]. What is Bob's optimal probability of guessing the value of X? For this problem, it is helpful to express the min-entropy in a different but equivalent form [79],

$$H_{\min}(X|B)_{\rho} = -\log \min_{\sigma} \{\mathrm{Tr}(\sigma) : \sigma \in \mathcal{P}(\mathcal{H}_B) \wedge \rho_{XB} \leq \mathbb{1}_X \otimes \sigma_B\}. \tag{3.19}$$

Expressed in this way, the minimisation corresponding to the quantity $2^{-H_{\min}(X|B)_{\rho}}$ is in exactly the right form for evaluation via an SDP. The dual problem is [80]

$$\max_{E_{XB}} \{\mathrm{Tr}(\rho_{XB} E_{XB}) : \mathrm{Tr}_X(E_{XB}) = \mathbb{1}_B\}, \tag{3.20}$$

where $E_{XB} \in \mathcal{P}(\mathcal{H}_{XB})$ and is classical on X. Slater's theorem [82] can be used to show strong duality, meaning the minimisation of the primal problem (Eq. (3.19)) is equal to the maximisation of the dual problem (Eq. (3.20)). Therefore, to evaluate $2^{-H_{\min}(X|B)_{\rho}}$ it suffices to perform the maximisation in Eq. (3.20).

Notice that since E_{XB} is non-negative and classical on X, we must have $E_{XB} = \sum_x |x\rangle\langle x| \otimes E_B^x$, meaning

$$\mathrm{Tr}(\rho_{XB} E_{XB}) = \sum_x p(x) \mathrm{Tr}(E_B^x \rho_B^x). \tag{3.21}$$

[3]Notice the difference in this setting, in which there is only a single copy of the state available, to the asymptotic IID setting, in which many copies of the same state are available.

Lastly, since $\mathrm{Tr}_X(E_{XB}) = \mathbb{1}_B$, the set $\{E_B^x\}_x$ must define a POVM on $\mathcal{H}_B$. Altogether, Eq. (3.20) can be rewritten as

$$2^{-H_{\min}(X|B)_\rho} = \max_{\{E_B^x\}_x} \sum_x p(x)\mathrm{Tr}(E_B^x \rho_B^x) := p_{\text{guess}}(X|B)_\rho. \tag{3.22}$$

The middle term is exactly Bob's probability of guessing X optimised over all possible measurements, hence the final equality. This interpretation will prove useful in the context of quantum USS schemes, since the adversary's goal is often to guess the signature (a classical string) using her stored correlated quantum information. However, for signatures, Eve normally only needs to guess a string that is approximately correct, so we modify these results in Chapter 6.

Private randomness extraction

Consider the scenario in which Alice and Bob have access to a *single* classical random variable X, and Eve holds side information E. Again, Alice and Bob want to publicly perform an operation on X to transform it into a shorter key Z, which is uniformly random and secret except with probability ϵ. What is the maximum size of Z, denoted $H_{\text{ext}}(X|E)_\rho$, that Alice and Bob are able to extract?

The optimal length of Z can be described in terms of the smooth min-entropy when Eve's information is either classical [64, 83] or quantum [67, 72]. In the case of Eve holding quantum side information

$$H_{\text{ext}}(X|E)_\rho \approx H_{\min}^\epsilon(X|E)_\rho \tag{3.23}$$

Similarly, the number of *perfectly* secret random bits that can be extracted is described by the (non-smooth) min-entropy.

One-shot CQSW theorem

Suppose Alice and Bob share the cq-state ρ_{XB} or, equivalently, that Alice holds the classical random variable X and Bob has access to quantum side information B. Alice wants to compress X to the classical message C so that, with access to B and C, Bob is able to perfectly reconstruct X except with probability ϵ. What is the minimum length, denoted $l_{\text{enc}}^\epsilon(X|B)_\rho$, of C for which this is achievable? Bounds on this length are given in terms of the smooth max-entropy as [84]

$$H_{\max}^{\sqrt{2\epsilon}}(X|B)_\rho \leq l_{\text{enc}}^\epsilon(X|B)_\rho \leq H_{\max}^{\epsilon_1}(X|B)_\rho + 2\log\frac{1}{\epsilon_2} + 4, \tag{3.24}$$

where $\epsilon_1, \epsilon_2 \geq 0$ such that $\epsilon = \epsilon_1 + \epsilon_2$. Similarly to the IID setting, this result is very useful in the context of QKD error correction.

One-way secret key distillation

For QKD, consider the overall state held by Alice, Bob and Eve following the transmission of the quantum states to Bob over the quantum channel. The state is the cqq-state

$$\rho_{XBE} = \sum_x p(x) \left|x\right\rangle \left\langle x\right|_X \otimes \rho^x_{BE}, \tag{3.25}$$

where Alice holds the X system, Bob holds the B system and Eve holds the E system. What is the length, denoted $l^\epsilon_{\text{secr}}(X;B|E)_\rho$, of the key that can be extracted, such that the key is a uniformly random string that is uncorrelated with E, but known to both Alice and Bob (with failure probability ϵ)? This corresponds to the achievable key generation rate in QKD when the adversary is allowed to perform the most general attacks allowed by quantum mechanics, i.e. coherent attacks. Upper and lower bounds are given in terms of the smooth min- and max-entropy as[4] [84]

$$\begin{aligned} l^{\epsilon+\epsilon'}_{\text{secr}}(X;B|E)_\rho &\geq H^{\epsilon'_1}_{\min}(X|E)_\rho - H^{\epsilon_1}_{\max}(X|B)_\rho - 4\log\frac{1}{\epsilon_2} - 3 \\ l^{\epsilon}_{\text{secr}}(X;B|E)_\rho &\leq H^{\sqrt{2\epsilon}}_{\min}(X|E)_\rho - H^{\sqrt{2\epsilon}}_{\max}(X|B)_\rho, \end{aligned} \tag{3.26}$$

where $\epsilon = \epsilon_1 + \epsilon_2 \geq 0$ and $\epsilon' = \epsilon'_1 + \epsilon_2 \geq 0$. The result can be intuitively understood as Alice extracting randomness independent of Eve at a rate of $H^{\epsilon'}_{\min}(X|E)_\rho$, as per the private randomness extraction subsection, but having to reduce this rate by $H^{\epsilon}_{\max}(X|B)_\rho$ to ensure Bob has zero uncertainty on the generated key, as per the one-shot CQSW theorem.

3.4.4 Useful results

Equipped with the definitions above, we are able to present two fundamental and widely applicable results that are used throughout quantum information theory. We will use these results later to prove Theorem 6.1, a result which forms the backbone of the security analysis of the quantum USS scheme presented in Chapter 6.

[4]This result can be improved via preprocessing performed by Alice on the X key. For simplicity we have neglected that possibility.

The data processing inequality

The data processing inequality (DPI) is an essential theorem in both classical and quantum information theory. Intuitively, the quantum DPI states that it is impossible to increase the information content of a state through local processing alone. Various different forms of the DPI hold for different entropy measures, but it can be generically stated as follows [69, 85].

Theorem 3.11 (Generic Data Processing Inequality)**.** *Let $\rho_{AB} \in \mathcal{D}(\mathcal{H}_A \otimes \mathcal{H}_B)$ and let $\mathcal{E}$ be a completely positive trace preserving (CPTP) map from B to B'. Set $\tau_{AB'} := \mathcal{E}(\rho_{AB})$. A generic entropy measure H satisfies the DPI if*

$$H(A|B')_\tau \geq H(A|B)_\rho. \tag{3.27}$$

The DPI holds for the classical Shannon entropy, the von Neumann entropy and the min- and max-entropies. Of particular importance to this thesis is the DPI for the smooth min-entropy which, under the same conditions as above, states that

$$H_{\min}^\epsilon(A|B')_\tau \geq H_{\min}^\epsilon(A|B)_\rho. \tag{3.28}$$

Proof. Let $\lambda = H_{\min}^\epsilon(A|B)_\rho$. By definition (c.f. (3.16) and (3.18)) there exists a state $\tilde{\rho}_{AB}$ that is ϵ-close to ρ_{AB} (in terms of the generalised purified distance), and a state $\sigma_B \in \mathcal{D}_{\leq}(\mathcal{H}_B)$, such that

$$\tilde{\rho}_{AB} \leq 2^{-\lambda} \mathbb{1}_A \otimes \sigma_B. \tag{3.29}$$

Rearranging, we see that the operator $2^{-\lambda}\mathbb{1}_A \otimes \sigma_B - \tilde{\rho}_{AB}$ is nonnegative. Since $\mathcal{E}$ is a linear CPTP map, this implies

$$\mathcal{E}\Big(2^{-\lambda}\mathbb{1}_A \otimes \sigma_B - \tilde{\rho}_{AB}\Big) \geq 0, \quad \text{and so} \quad \mathcal{E}(\tilde{\rho}_{AB}) \leq 2^{-\lambda}\mathbb{1}_A \otimes \mathcal{E}(\sigma_B). \tag{3.30}$$

Therefore, if we can show that (i) $\mathcal{E}(\tilde{\rho}_{AB})$ is ϵ-close to $\tau_{AB'}$ and (ii) that $\mathcal{E}(\sigma_B) \in \mathcal{D}_{\leq}(\mathcal{H}_{B'})$ then by the definition of smooth min-entropy we will have shown that $H_{\min}^\epsilon(A|B')_\tau \geq \lambda$, thus proving the DPI. The proof of (ii) follows immediately from the fact that $\mathcal{E}$ is a CPTP map. To prove (i) we employ a useful property of the generalised purified distance, namely, the monotonicity property [65] which states that if two sub-normalised states are ϵ-close, then applying a CPTP map to each

state can only decrease the distance between them. In our case

$$P(\rho_{AB}, \tilde{\rho}_{AB}) \leq \epsilon \Rightarrow P(\tau_{AB'}, \mathcal{E}(\tilde{\rho}_{AB})) \leq \epsilon. \tag{3.31}$$

□

Entropic uncertainty relations

The uncertainty principle states that there is an unavoidable uncertainty in the measurement outcomes of non-commuting observables. The standard Robertson relation [3] expresses the uncertainty of the measurement outcomes of two observables, acting on a particular state, in terms of the standard deviation. The form of this uncertainty relation is undesirable for two main reasons. First, the uncertainty depends not only on the observables, but also on the particular state on which the observables are measured. Often one wants to know the degree to which two observables are incompatible, without reference to any particular state. Second, the standard deviation as a measure of uncertainty has no clear operational interpretation, and so is less useful than quantifying uncertainty in terms of an entropic quantity with a clear operational meaning [69, 79].

Entropic uncertainty relations remedy both of these issues. The first entropic uncertainty relation was provided for the position and momentum observables in 1957 by Hirschman [86]. Later, in 1983 Deutsch highlighted the advantage of inequalities that are state-independent, and provided the first general entropic uncertainty relation holding for any two non-degenerate observables [87]. Of particular importance to this thesis is a generalisation of the entropic uncertainty relation proposed by Deutsch to the smooth min- and max- entropy.

Consider any tripartite state $\rho_{ABC} \in \mathcal{D}(\mathcal{H}_A \otimes \mathcal{H}_B \otimes \mathcal{H}_C)$ and any two POVMs, $\mathcal{X} = \{M_x\}_x$ and $\mathcal{Z} = \{N_z\}_z$, acting on system A. Following a measurement of A with respect to $\mathcal{X}$, define the reduced state of ρ_{ABC} (tracing out system C) to be

$$\rho_{XB} := \sum_x |x\rangle\langle x| \otimes \tau_B^x, \quad \text{where} \quad \tau_B^x = \mathrm{Tr}_{AC}(M_x \rho_{ABC}). \tag{3.32}$$

Similarly, define ρ_{ZC} as the reduced state of the system following a measurement of $\mathcal{Z}$ on A, with system B traced out. Lastly, define

$$q := \log \frac{1}{\max_{x,z} ||\sqrt{M_x}\sqrt{N_z}||_\infty^2}, \tag{3.33}$$

where $||\,.\,||_\infty$ is the spectral norm, or the Schatten ∞-norm, which in our case can

be evaluated as the largest singular value of a matrix.

Theorem 3.12 ([88]). *Let $\epsilon \geq 0$, and let $\rho_{ABC}, \mathcal{X}, \mathcal{Z}, \rho_{XB}, \rho_{ZC}$ and q be defined as above. Then*

$$H_{\min}^{\epsilon}(X|B)_{\rho_{XB}} + H_{\max}^{\epsilon}(Z|C)_{\rho_{ZC}} \geq q. \tag{3.34}$$

This theorem quantifies the level of incompatibility of two measurements in terms of the quantity q. It shows there is an unavoidable trade off to be made – an increase in one's ability to predict the outcome of measurement $\mathcal{X}$ is necessarily associated with a decrease in one's ability to predict the outcome of measurement $\mathcal{Z}$, and vice versa. Entropic uncertainty relations have become a powerful tool in many areas of quantum cryptography and, for example, the above relation has been used to provide a simple and elegant unconditional security proof for QKD [89]. Despite this, entropic uncertainty relations are only understood in very limited settings and many fascinating open questions remain [90].

3.5 Decoy state QKD

The QKD protocol described in Section 3.3.1 employs single-photon states to encode and transmit information. Though convenient for the theoretical analysis, perfect single-photon sources do not exist. Further, it can often be more convenient to use coherent state sources due to the relative maturity of the technology when compared to single-photon sources.

As such, phase-randomised coherent sources are widely used as an alternative to single-photon sources. These sources emit light as a classical mixture of number states [91]

$$e^{-|\alpha|^2} \sum_{k=0}^{\infty} \frac{|\alpha|^{2k}}{k!} |k\rangle \langle k| , \tag{3.35}$$

where $|\alpha|^2$ is the mean photon number of the pulse. If the mean photon number is chosen to be small, the majority of pulses will contain either 0 or 1 photon.

The exact number of photons in each pulse emitted by Alice's source is not directly observable by Alice or Bob. Nevertheless, when the pulse contains zero photons neither Eve nor Bob receive any information. Conditioned on the pulse containing exactly one photon, the state is exactly the single-photon state considered by the normal theoretical analysis and so security is guaranteed. Problems arise when the pulse contains more than one photon, in which case powerful photon number splitting (PNS) attacks can be employed by Eve to gain full information on

the state sent by Alice while causing no errors. Essentially, for a pulse containing $k > 1$ photons, Eve can siphon off one of the photons and store it in a quantum memory. She forwards all remaining photons and measures her stored state only after the public basis announcements are made, thereby discovering the identity of the state with certainty and causing zero additional disturbance. Even worse, Eve can measure the number of photons contained in each pulse and selectively suppress the single-photon states, thereby increasing the fraction of states reaching Bob originating from multi-photon pulses. Since Eve can completely control the channel losses, by decreasing the loss rates on multi-photon pulses she is able to at least partially offset the additional losses caused by suppressing single-photon pulses.

Despite this, protocols using phase-randomised coherent sources (and without the decoy-state technique [92]) can still achieve unconditional security by reducing the mean photon number of each pulse so that multi-photon events are rare. Worst-case estimates can then be used to bound Eve's potential information, even if Eve performs the strategies outlined above (or any other strategy). The secret key rate S, per pulse sent by Alice, can be expressed in terms of: Q_α and E_α, the count rate and quantum bit error rate (QBER) of the signal states, respectively; and Ω and e_1, the fraction and QBER of Bob's detection events originating from single-photon pulses, respectively. Ref. [93] finds

$$S \geq Q_\alpha(-h(E_\alpha) + \Omega[\, 1 - h(e_1)\,]). \tag{3.36}$$

Q_α and E_α do not depend on the photon number and are easy to estimate directly from the experimental data. Ω and e_1 are harder to estimate, and the worst-case estimates proposed in Ref. [93] were too conservative, leading to an unacceptably large drop in the key generation rates.

To solve this problem, decoy-state QKD was proposed in order to find more accurate estimates of Ω and e_1. Let Y_k be the conditional probability that Bob detects a signal, given that a k-photon pulse is emitted by Alice. The count rate can be expressed as

$$Q_\alpha = Y_0 e^{-\alpha} + Y_1 e^{-\alpha}\alpha + \cdots + Y_k e^{-\alpha}(\alpha^k/k!) + \ldots. \tag{3.37}$$

Similarly, if we define e_k to be the QBER arising from k-photon signal pulses, then

$$Q_\alpha E_\alpha = Y_0 e^{-\alpha} e_0 + Y_1 e^{-\alpha}\alpha e_1 + \cdots + Y_k e^{-\alpha}(\alpha^k/k!)e_n + \ldots. \tag{3.38}$$

The essential insight of decoy-state QKD is that Y_k and e_k do not depend on the intensity level of the pulses. Therefore, since Q_α and E_α can be observed easily for all values of α, the above two equations specify infinitely many linear equations in Y_k and e_k. If Alice was to choose infinitely many different intensities to transmit, Alice and Bob could accurately estimate Y_k and e_k for all values of k using the equations above. If any of the estimated parameters are found to be significantly different to those expected from the channel, the protocol is aborted. This additional testing stage severely restricts the strategies available to Eve.

Looking back at Eq. (3.36), the only parameters we actually need to estimate are e_1 and Y_1 (since it can be shown that $\Omega = Y_1 \mu e^{-\mu}/Q_\mu$, where μ is the intensity chosen for signal states). Therefore Alice does not need to choose infinitely many different intensities. Instead, using a signal intensity and just two "decoy" intensities allows us to find an accurate lower bound on Ω and an accurate upper bound on e_1. This means we are able to say with confidence that Eve has not suppressed more than an insignificant number of single-photon states, and we also know Bob's error rate on single-photon pulses.

The experimental benefits of decoy-state QKD are enormous. It allows experimentalists to use mature laser technologies as a light source, and to use signal pulses with a relatively large mean photon number ($\alpha = O(1)$) while still maintaining security.

3.6 Classical authentication

Man-in-the-middle attacks are powerful strategies in which the adversary intercepts communications between two legitimate parties and resends an altered message pretending that it originated with the legitimate sender. Without protection against these attacks all cryptographic protocols, including QKD, would be insecure. Classical authentication schemes using message authentication codes (MACs) are one way of eliminating this type of attack. A MAC is a two-party protocol used by honest participants, Alice and Bob, to authenticate the contents of a message that is sent. To authenticate the message, Alice appends a tag (the authentication code) that depends on the contents of the message. Bob is able to check that the tag is correct based on the contents of the message. To provide unconditional security, MACs require the two legitimate communicating party's (Alice and Bob) to share a secret key. For this reason, quantum key distribution should more accurately be

called quantum key expansion[5].

MACs will be useful to us in Chapter 8 in which we use them to create a particularly efficient classical USS scheme. At first glance, MACs may seem similar to signatures since they both aim to provide authentication of messages. However, there are significant differences in both the problem setting and the protocol aims. The most obvious difference is in the setting: MACs, formally defined below, are a two-party protocol in which both the sender and receiver are assumed to be honest; USS schemes are an $N \geq 3$ party protocol in which any participant could be dishonest. Another difference is the protocol aims. As stated in Section 2.3.1, signatures aim to provide unforgeability, transferability and non-repudiation of messages. Transferability is a requirement with no MAC analogue, since MACs are a two-party protocol. As we shall see, even if one considers external parties, MAC schemes are not designed to provide transferability since they are symmetric key schemes. More concretely, both Alice and Bob have the ability to send authenticated messages since both have full access to the secret key. Therefore, a message can be authenticated as having originated from either Alice or Bob, but there is no way of proving to an outsider exactly which party sent the message. This is fine for the MAC setting, in which two honest parties want to authenticate messages between themselves only. For signatures on the other hand, where there are N mutually distrustful participants, this would not be sufficient to provide either unforgeability or non-repudiation, since all participants with access to the key could produce "authenticated" messages, and MACs provide no mechanism to distinguish exactly who sent the message.

3.6.1 Message authentication codes

To use a MAC, Alice and Bob must choose a secret key, k, in advance of their communication. When Alice wants to send a message m, she computes a tag t from the message and the secret key k. She sends (m, t) to Bob who, using m and k, is able to verify whether t is the correct tag for the given message. Without access to k, an adversary cannot find the correct tag for a different message m', and so cannot alter the message in an undetectable way.

Definition 3.13 (Message authentication codes [94])**.** A MAC is defined by three algorithms (Gen, Mac, Ver) such that:

[5]Importantly, the secret key required for unconditionally secure authentication is small, and QKD generates more secret key than it consumes meaning it is still a very useful protocol. In fact, if we assume the existence of a small initial shared secret key (used to authenticate the classical channels), QKD can be used to expand it *arbitrarily*.

1. The key generation algorithm Gen takes the input security parameter and outputs a uniform random key $k \in \mathcal{K}$, where $\mathcal{K}$ is the key set.

2. The message authentication code generation algorithm Mac takes as input the key k and a message $m \in \mathcal{M}$, and outputs the tag $t \in \mathcal{T}$, where $\mathcal{M}$ and $\mathcal{T}$ are the message and tag set respectively.

3. The verification algorithm Ver takes as input the key k, a message m and a tag t and outputs either 1 meaning valid or 0 meaning $\mathsf{invalid}$.

Since it is always possible for the adversary to randomly guess the tag, the highest security level we can hope to achieve is that the adversary cannot guess a valid tag except with probability $1/|\mathcal{T}|$. This security level is achievable in the information-theoretic security setting, but only if we restrict the number of messages authenticated using the MAC. We consider the case of a one-time MAC, whereby Alice and Bob use the scheme to authenticate only a single message.

Definition 3.14 (One-time MAC)**.** A one-time MAC is ϵ-secure if, given access to a message-tag pair of her choosing, (m, t), the adversary cannot output a message tag pair, (m', t') with $m \neq m'$, such that

$$\mathsf{Ver}_k(m', t') = 1, \tag{3.39}$$

except with probability ϵ.

This definition means that, even when provided with a valid message-tag pair of her choice, the adversary cannot substitute in a distinct message with a valid tag, except with probability ϵ.

3.6.2 Strongly universal functions.

In this section we consider how to construct a MAC. To be secure, the MAC must be such that knowledge of a single message-tag pair gives almost no information on the tag of any distinct message, or, in other words, distinct message-tag pairs should be essentially independent. Suppose Mac was simply a function chosen completely at random from the set $\mathcal{R} = \{f : \mathcal{M} \to \mathcal{T}\}$, i.e. $\mathsf{Mac} = f_k(m)$, where the secret key k is used as an index to specify which function is chosen from $\mathcal{R}$. In this case, the protocol would proceed as follows:

1. Alice and Bob would agree in advance on a secret key k, chosen uniformly at random, specifying a function $f_k \in \mathcal{R}$.

2. To communicate an authenticated message m, Alice sends (m, t), where $t = f_k(m)$ is the message tag.

3. To verify the message, Bob checks that $t = f_k(m)$. Bob rejects the message if $t \neq f_k(m)$.

A MAC generated in this way would clearly satisfy the tag independence property. An adversary with no knowledge of k would gain no benefit from holding a valid pair (m, t) when trying to generate the correct tag for $m' \neq m$. In fact, even knowing (m, t) for *all* values of $m \neq m'$ would not help the adversary to find the correct tag for m'. The problem with this construction is that to specify a completely random function requires a key exponentially larger than the message being authenticated (there are 2^{2^n} Boolean functions with input size n), meaning Alice and Bob would need to share a key exponentially larger than the message being signed! Clearly, this would be extremely impractical. For this purpose, we introduce the notion of a strongly universal function.

Definition 3.15 (Strongly universal)**.** A set of functions $\mathcal{H} = \{h : \mathcal{M} \to \mathcal{T}\}$ is strongly universal (SU) if for all distinct $m, m' \in \mathcal{M}$ and for all $t, t' \in \mathcal{T}$

$$|\{h \in \mathcal{H} : h(m) = t \wedge h(m') = t'\}| = \frac{|\mathcal{H}|}{|\mathcal{T}|^2} \,. \tag{3.40}$$

The meaning of this definition is that, even after the adversary has seen a single message-tag pair, the tag for any other message is uniformly distributed across $\mathcal{T}$, and so the adversary can do no better than to randomly guess the tag for any distinct message. In this case the scheme achieves the maximum security level of $1/|\mathcal{T}|$. In this sense, a function chosen at random from a SU set is indistinguishable from a truly random function given only a single input/output pair, i.e. generating the MAC using a function selected from a SU set provides the same security as using a truly random function, *given* that the MAC is used to authenticate only a single message. The benefit of using SU sets is that the key needed to specify a function in the set is much smaller than the key needed to specify a truly random function. Nevertheless, to specify a SU function still requires a number of bits approximately equal to the size of the message being authenticated (see for example the construction in [95]). Is it possible to do any better? Fortunately, the answer is yes.

3.6.3 Almost strongly universal functions

By relaxing the requirements of SU sets, we are able to find much smaller sets that still approximate truly random functions when used only once.

Definition 3.16 (Almost strongly universal functions [96])**.** A set of functions $\mathcal{H} = \{h : \mathcal{M} \to \mathcal{T}\}$ is ϵ-almost strongly universal (ϵ-ASU$_2$) if for all distinct $m, m' \in \mathcal{M}$ and for all $t, t' \in \mathcal{T}$

1. $|\{h \in \mathcal{H} : h(m) = t\}| = |\mathcal{H}|/|\mathcal{T}|$,
2. $|\{h \in \mathcal{H} : h(m) = t \wedge h(m') = t'\}| \leq \epsilon \frac{|\mathcal{H}|}{|\mathcal{T}|}$.

The meaning of this definition is that, after seeing a single message-tag pair, the adversary gains *almost* no information on the identity of distinct message tags, and so cannot do *much* better than to randomly guess the tag of a distinct message. Note that the cost of increased efficiency (i.e. k being smaller) is that security is no longer perfect. The probability of the adversary guessing a correct tag when Mac uses an ϵ-ASU$_2$ set is at most $\epsilon \geq 1/|\mathcal{T}|$. ASU$_2$ sets were first introduced in Ref. [31], in which the authors choose $\epsilon = 2/|\mathcal{T}|$ and propose an ϵ-ASU$_2$ set built from polynomials over finite fields. In this construction a key size of approximately $4 \log |\mathcal{T}| \log \log |\mathcal{M}|$ is required to specify a function within the set, i.e. the key size is approximately logarithmic in the bit size of the message being authenticated. Given a single message-tag pair of her choice, the adversary is only able to guess the tag of a distinct message with probability $\epsilon = 2/|\mathcal{T}|$. Therefore, just as before, the protocol can be made arbitrarily secure by increasing the length of the tag. Since the original Wegman-Carter construction, many other ϵ-ASU$_2$ sets have been proposed [96–100], each tailored to provide different benefits depending on the desired application[6].

Application to signatures

In Chapter 8 we modify the generic MAC protocol using ϵ-ASU$_2$ functions to make it suitable for use in the USS setting. Essentially, for the case of a single sender and many possible recipients, we provide each sender-recipient pair with multiple keys specifying independently chosen functions from an ϵ-ASU$_2$ set. Each recipient distributes a selection of his keys to all other recipients in a manner similar to the secret sharing scheme introduced by Chaum, Crépeau and Damgard [101]. This distribution effectively breaks the sender/receiver symmetry and provides all recipients

[6]For example, different constructions place different levels of importance on features such as key length, tag length, ease of computability, etc.

with partial information on the overall key held by the sender – enough to verify the tags (the signature), but not enough to reproduce them. In this way, we are able to maintain the efficiency of unconditionally secure MACs while also guaranteeing the unforgeability and transferability of messages. Full details are provided in Chapter 8.

3.7 Oblivious transfer

Oblivious transfer (OT) is one of the most important and well known primitives in modern cryptography. Its prominence stems from the fact that it can be used as the foundation for all secure two-party computations – with OT, all secure two-party computations are possible [102, 103]. OT comes in many different flavours, but in this thesis we consider only information-theoretically secure stand-alone protocols for 1-out-of-2 OT (1-2 OT). Formal definitions of 1-2 OT are provided in Chapter 9.

Informally, 1-2 OT is a two-party protocol in which there is a sender who aims to provide a receiver with exactly one out of two possible messages, such that the receiver chooses which message to receive. In other words, Alice inputs two bits, x_0 and x_1, and Bob inputs a single bit, b. The protocol outputs x_b to Bob with the guarantees that Alice does not know b, and that Bob does not know x_{1-b}. A dishonest Alice aims to find the value of b, and her optimal probability of doing so is denoted A_{OT}. A dishonest Bob aims to correctly guess both x_0 and x_1, and his optimal probability of doing so is B_{OT}. Ideal 1-2 OT, in which $A_{OT} = B_{OT} = 1/2$, is known to be impossible in the information-theoretic setting [104, 105].

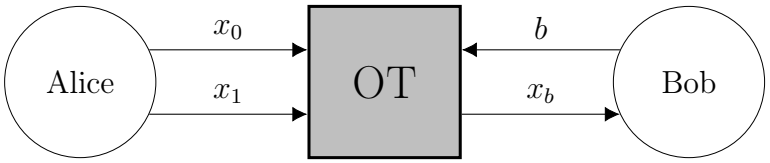


Figure 3.1: 1-out-of-2 oblivious transfer. Alice has two inputs, x_0 and x_1, and receives no outputs. Bob has one input, b, and receives one output x_b.

Relation to signatures

Oblivious transfer is a two-party protocol performed between mutually distrustful participants. It is therefore fundamentally different to USS schemes which always contain $N \geq 3$ participants. However, the distribution stage of USS schemes is often

reminiscent of OT. For example, consider the distribution stage of the three-party USS scheme P2 described in Ref. [1]. It proceeds as follows:

Distribution Stage V1

1: For each possible future message $m = 0$ and 1, Alice selects two n-bit keys, A_B^m and A_C^m.
2: She uses a secret channel to send A_B^m to Bob and A_C^m to Charlie.
3: Bob splits his key into two equally sized sets, B_1^m and B_2^m, and sends B_2^m to Charlie using a secret channel. Similarly, Charlie creates C_1^m and C_2^m, and forwards C_2^m to Bob.

To sign message m Alice's signature is the $2n$-bit string formed from the concatenation of A_B^m and A_C^m. If Alice sends the message to, say, Bob, he checks her signature against both B_1^m and C_2^m independently, and records the number of mismatches. The message is accepted if both sets contain fewer than a threshold rate (s_a) of mismatches with Alice's signature. To forward the message, Bob forwards to Charlie exactly what he received from Alice. Charlie checks the message similarly to Bob, but uses C_1^m and B_2^m and the verification threshold $s_v > s_a$. The protocol is secure for two reasons.

1. *Security against forging.* Bob cannot learn the bit values in C_1^m, which means he is unable to produce a valid signature.

2. *Security against repudiation and transferability.* For each position in Alice's signature, she is completely unaware whether Bob (or Charlie) will be checking that bit, i.e. each bit of her signature has a probability of 1/2 of being checked by Bob and a probability of 1/2 of being checked by Charlie. This means Alice cannot bias a signature to contain more errors for Bob than for Charlie (or vice versa).

These security guarantees are very similar to those required by 1-2 OT – namely, Bob cannot discover all of Alice's inputs (i.e. A_B^m and A_C^m) and Alice cannot discover any of Bob's inputs (i.e. the index positions of B_1^m and C_2^m). To make the connection more explicit, suppose the participants had access to a black-box implementation of (possibly imperfect) 1-2 OT with cheating parameters A_{OT} and B_{OT}. Then the distribution stage of P2 might proceed as stated below (note that the protocol below is not meant as a full signature protocol, and serves only to highlight how imperfect OT *might* be applied to signatures. A rigorous security analysis of a fully stated scheme would be necessary before it could be claimed secure).

Distribution Stage V2 Sketch

1: For each possible future message $m = 0$ and 1, Alice selects two n-bit keys, A_B^m and A_C^m.
2: Bob and Charlie both uniformly at random select the n-bit strings B and C, respectively.
3: For each $i = 1, \ldots, n$, Alice and Bob use the black-box implementation of 1-2 OT. Alice's inputs are the i'th bits of A_B^m and A_C^m and Bob's input is the i'th bit of B. Alice and Charlie do the same, except Charlie uses C to specify his inputs.
4: Bob and Charlie sacrifice a small portion of their outputs to ensure Alice's inputs are the same for each of them.

A protocol of this form would remove the need for the exchange process between Bob and Charlie seen in Step 3 of the distribution stage V1. The security guarantees of the black-box 1-2 OT imply that Bob can guess at most a fraction B_{OT} of Alice's inputs. As long as $B_{OT} < 1$, Bob cannot perfectly reproduce Alice's signature and so the protocol should be secure against forging.

On Alice's side, the protocol ensures that she can only correctly guess a fraction A_{OT} of the indices in B and C. This is more problematic, since the gap between the authentication and verification thresholds (s_a and s_v) is often small. By guessing the indices held by each participant, Alice can bias the expected error rate so that one participant has an expected error rate $A_{OT} - 1/2$ higher than the other[7]. As such, if imperfect OT schemes are to be used in constructing USS schemes, then the security guarantees on the sender (Alice) must be stricter than the security guarantees on the receiver.

Direct vs indirect 1-2 OT

It is not surprising that USS schemes can be created using 1-2 OT, since 1-2 OT can be used to perform *any* secure multiparty computation. However, as stated above, perfect 1-2 OT is known to be impossible in the information-theoretic setting. Precisely what security parameters are attainable for imperfect protocols remains an interesting open question (which we consider in depth in Chapter 9), but it is known that [106]

$$\max\{A_{OT}, B_{OT}\} \geq 2/3. \tag{3.41}$$

The distribution stage V1 can be thought of as allowing Alice and Bob (and Alice and Charlie) to effectively perform an imperfect version of OT in which $A_{OT} = 1/2$

[7]For example, if $A_{OT} = 1$, Alice knows exactly which bits were selected by Bob and Charlie. Therefore, for the cases when Bob selected a different bit to Charlie (which happens with probability 1/2), Alice can ensure that her signature matches Bob's chosen bit, but not Charlie's. In this way, if Bob sees an error rate of e_B, then Charlie will see an error rate of $1/2 + e_B$.

(since Alice has no information on which bits Bob kept) and $B_{OT} = 3/4$ (since Bob learns 3/4 of Alice's bit values). However, there are also results (again contained in Ref. [106]) which show that these cheating parameters are impossible to achieve for 1-2 OT in the standard information-theoretic setting. The reason that Alice and Bob are able to beat the known impossibility bounds is that they use an untrusted third party (Charlie) as an *additional* resource, over and above the resources normally available in the two-party information-theoretic setting. In other words, in the distribution stage V1, Alice and Bob can be thought of as performing OT indirectly using Charlie as a facilitator.

Nevertheless, the use of Charlie as an additional resource has one main disadvantage: it requires a high level of recipient-recipient interaction. Especially for USS schemes with larger numbers of protocol participants, the additional pairwise interactions required to facilitate this indirect form of OT between each sender-recipient pair become onerous and reduce the viability of the protocol. Accordingly, it might instead be desirable to perform direct OT, as in the distribution stage V2. In these schemes, it may be possible for the majority of communications to take place only between the sender and each receiver. The attainable cheating probabilities A_{OT} and B_{OT} will be worse than for indirect OT schemes, but may still be sufficient to construct a secure USS scheme.

In Chapter 9 we explore direct implementations of imperfect 1-2 OT, and derive new bounds on the attainable cheating probabilities in the standard information-theoretic setting. Our results have the advantage of parametrising A_{OT} and B_{OT} in terms of a single variable, thereby allowing us to derive bounds on B_{OT} when A_{OT} is kept close to 1/2, as seems necessary for USS schemes.

3.8 Byzantine agreement

Byzantine agreement is a problem that has found many applications in fault-tolerant distributed computing systems. Its name is derived from an analogy with a Byzantine army laying siege to a city [54]. The army contains multiple detachments: one led by the commanding general (the sender), and all others led by subordinate generals (the recipients). The commanding general is supposed to coordinate his army and give an order of either "attack" or "retreat". Importantly, any number of the generals, including the commanding general, could in fact be traitors (dishonest). The generals can only communicate pairwise via messenger (the messengers are always honest), and they want to devise a system such that:

1. All loyal generals agree on a common plan of action.

2. If the commanding general is loyal, then all loyal generals agree on the commanding general's plan.

For computing applications, the commanding general corresponds to a single node transmitting a message, and the subordinate generals are simply the other nodes contained in the network. Finding a protocol which satisfies the above two points would mean that, for distributed computing applications, correct nodes would be able to work together coherently even in the presence of faulty nodes. More precisely, Byzantine agreement is defined as follows.

Definition 3.17 (Byzantine agreement [107])**.** Byzantine agreement is a protocol with a sender and N receivers such that any number of the participants can be dishonest. The sender chooses an input value $x \in \{0, 1\}$ and the receivers must decide on an output value. The protocol should ensure that all honest receivers agree on the *same* output value $y \in \{0, 1\}$. Additionally, if the sender is honest, the protocol should ensure that $y = x$.

In the standard setting, all participants are connected pairwise via authenticated classical channels. As can be seen from the definition above, a more descriptive name for the Byzantine agreement problem is *authenticated broadcast*. Authenticated broadcast resembles USS schemes in many ways: both protocols aim to send messages that honest participants will accept and agree on; both are multiparty protocols containing an adversary who works within the scheme; and neither scheme aims for secrecy. The similarities are such that *if* participants are all able to sign messages with unconditionally security, then a scheme for authenticated broadcast exists no matter how many participants are dishonest [54], i.e.

$$\text{Ability to sign messages} \Rightarrow \text{Authenticated broadcast.} \tag{3.42}$$

However, there are differences between USS schemes and authenticated broadcast, with the two main ones being:

1. In authenticated broadcast all recipients interact to agree on the validity of a message. For USS schemes on the other hand, the sender sends a message only to a single recipient, who can non-interactively check the validity of the contents and be guaranteed to be able to transfer the message a finite number of times. Transferability does not make sense in the context of broadcast.

2. Authenticated broadcast does not allow participants to abort the protocol, whereas abort is a valid option in USS schemes.

These differences are significant, and lead to different impossibility results in the two schemes. For example, authenticated broadcast between three participants is impossible to achieve unless all participants are honest. More generally, authenticated broadcast is impossible unless $t < 1/3$ of the participants are dishonest [54]. On the other hand, USS schemes are possible between three participants even in the presence of a single dishonest participant. Immediately then, we see that there are settings in which USS schemes are possible yet authenticated broadcast is not. At first glance, this statement seems to contradict (3.42). The reason there is in fact no contradiction is that USS schemes allow an abort option, whereas authenticated broadcast does not. Therefore, although an USS may exist, this only implies that participants can *either* sign a message with unconditional security, *or* about the protocol. Nevertheless, the existence of a USS scheme does allow for a very similar version of broadcast, known as *detectable broadcast*.

Definition 3.18 (Detectable broadcast [107])**.** Detectable broadcast is a protocol with a sender and N receivers such that any number of the participants can be dishonest. The sender chooses an input value $x \in \{0, 1\}$ and the receivers must decide on an output value *or* abort the protocol. If all participants are honest, the protocol should achieve authenticated broadcast. Otherwise, the protocol should either achieve authenticated broadcast or have all honest players abort.

Detectable broadcast, though weaker than authenticated broadcast, is still powerful enough for many applications. Importantly, the existence of a USS scheme allows participants to perform detectable broadcast, i.e.

$$\text{USS schemes} \Rightarrow \text{Detectable broadcast.} \tag{3.43}$$

The converse implication does not hold in general[8].

In this subsection we introduced Byzantine agreement (authenticated broadcast) and discussed its relevance to USS schemes. We found that USS schemes are more closely related to a slightly weaker notion of broadcast, namely, detectable broadcast, which allows participants the option of aborting if necessary. The existence of a USS scheme implies the ability for the participants to perform detectable broadcast,

[8]For example, if $2N$ participants are connected by a detecable broadcast channel, and $N + 1$ participants are dishonest, they can always cheat in the protocol simply by forcing the majority vote dispute resolution process.

but *not* authenticated broadcast. Since authenticated broadcast is an expensive resource, and stronger than detectable broadcast, in this thesis we have avoided USS schemes which assume authenticated broadcast as a resource, such as those presented in Refs. [37, 39].

Chapter 4

USS security framework

4.1 Introduction

In Chapter 2 we introduced and motivated the concept of USS schemes. Our definitions were colloquial and aimed to provide an insight into the spirit of signatures rather than set out their formal requirements. Nevertheless, rigorous security definitions are essential to modern cryptography, and are a basic requirement for the study of any cryptographic protocol [94]. Their use facilitates a more logical research strategy and helps to distill the true goals of a scheme. This structured approach also helps to evaluate and compare different schemes, stripping away superfluous or unnecessary features. The definitions stated in this chapter are taken from the quantum USS security framework proposed by Arrazola et. al in Ref. [34], which generalises the security model existing for classical USS schemes proposed in Ref. [33].

4.2 USS schemes

To specify a security model, we must first define what an USS scheme is and the core elements it should contain.

Definition 4.1 (Unconditionally Secure Signature Scheme)**.** An USS scheme $\mathcal{Q}$ is an ordered set $\{\mathcal{P}, \mathcal{M}, \Sigma, L, \mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver}\}$ where

- The set $\mathcal{P} = \{P_0, P_1, ..., P_N\}$, is the set containing the signer, P_0, and the N potential receivers.
- $\mathcal{M}$ is the set of possible messages.
- Σ is the set of possible signatures.

- Gen is the generation algorithm that gives rise to the functions Sign and Ver, used respectively to generate a signature and verify its validity. More precisely, the generation algorithm specifies the instructions for the communication that takes place in the distribution stage of the protocol. Based on the data obtained during the distribution stage, the generation algorithm instructs how to construct the functions Sign and Ver. The generation algorithm includes the option of outputting an instruction to abort the protocol.

- Sign: $\mathcal{M} \to \Sigma$ is a deterministic function that takes a message $m \in \mathcal{M}$ and outputs a signature $\sigma \in \Sigma$.

- $L = \{-1, 0, 1, ..., l_{\max}\}$ is the set of possible verification levels of a signed message. A verification level l corresponds to the minimum number of times that a signed message can be transferred sequentially to other recipients. For a given protocol, the maximum number of sequential transfers that can be guaranteed is denoted by $l_{\max} \leq N$.

- Ver: $\mathcal{M} \times \Sigma \times \mathcal{P} \times L \to \{\mathsf{True}, \mathsf{False}\}$ is a deterministic function that takes a message m, a signature σ, a participant P_i and a level l, and gives a boolean value depending on whether participant P_i accepts the signature as valid at the verification level l.

Notation 4.2. For a fixed participant, P_i, at a fixed verification level, l, we denote the verification function as $\mathsf{Ver}_{i,l}(m, \sigma) := \mathsf{Ver}(m, \sigma, i, l)$.

Notation 4.3. A signature σ on a message m is called i-acceptable if $\mathsf{Ver}_{i,0}(m, \sigma) = \mathsf{True}$, i.e. (m, σ) is i-acceptable if participant P_i will accept (m, σ) as a valid message-signature pair at the lowest verification level, $l = 0$.

Although the signing and verification algorithms are deterministic functions, the generation algorithm (which creates them) must include randomness for the protocol to be secure. The inclusion of randomness means that an adversary will not have a full specification of the signing and verification algorithms held by each recipient, a fact that is crucial for preventing dishonesty. The randomness could be generated in a variety of ways. For example, in many quantum USS schemes it is generated via the inherent randomness of quantum measurement outcomes.

Correctness

The definition above describes the core elements that must be present in any USS scheme. However, it does not provide any information on what these elements are

supposed to achieve. An integral part of the definition of USS schemes is the specification of the functionality that an USS scheme must possess. A scheme providing this functionality is said to be working correctly.

Definition 4.4 (Correctness of USS schemes)**.** An USS protocol $\mathcal{Q}$ is correct if $\mathsf{Ver}_{i,l}(m, \mathsf{Sign}(m)) = \mathsf{True}$ for all $m \in \mathcal{M}$, $i \in \{1, ..., N\}$, and $l \in L$.

Definition 4.5 (ϵ-correct)**.** An USS protocol is called ϵ-correct if it is correct except with probability ϵ.

These definitions formalise the intuitive notion of what an USS scheme is – a correct USS scheme is one in which the signing algorithm produces signatures that will be accepted by the verification algorithms. As such, in the absence of errors, all signatures created by the signing algorithm of a correct USS scheme will be accepted as valid by all verification algorithms. Of course, in reality not all protocol participants may be honest. Therefore, as well as correctness, USS schemes must also be secure. Security definitions are provided in Section 4.4 after we have formally introduced the process of dispute resolution.

Verification levels

We also briefly discuss the notion of *verification levels* since they have no analogue in public-key digital signature schemes and are perhaps confusing. Intuitively, if a signature is found to be valid, then all participants should agree on its validity. This is the case for all real-world signature schemes with the "universal verifiability" property, but unfortunately this is not possible when one wants unconditional security in the standard resource model. Instead, the best that can be done is to provide guarantees on how many times a message can be forwarded in sequence and accepted as valid[1]. Nevertheless, this is sufficient for many applications.

As a standard example consider a protocol involving a signer, Alice, a receiver, Bob, his local bank branch, and the bank's headquarters. Upon receiving a cheque from Alice, Bob wants to deposit the cheque into his bank account. However, his local branch will only credit Bob's account once the cheque has also been accepted by their headquarters. Therefore, for the cheque to be useful to Bob it must be sequentially transferable at least twice (i.e. from Bob to his local branch and from the local branch to the headquarters).

[1]As we shall see, even if the message is only *guaranteed* to be transferable l times, it is still possible that it can be securely transferred many more than l times. Indeed, if participants are honest, it is likely that the message will actually be able to be transferred N times, but there are no a priori guarantees that the message is N-transferable. As such, from an efficiency point of view finite transferability is not necessarily a limiting factor.

Definition 4.6 (l-transferable)**.** A signature σ on a message m is l-transferable if $\mathsf{Ver}_{(i,l)}(m,\sigma) = \mathsf{True}$ for all $i \in \{1,\ldots,N\}$ and there exists j such that $\mathsf{Ver}_{(j,l+1)}(m,\sigma) = \mathsf{False}$. For $l = l_{\max}$, the function $\mathsf{Ver}_{(j,l_{\max}+1)}(m,\sigma)$ is not defined and we assume by convention that it is always False.

In other words, a message is l-transferable if l is the highest level at which *all* participants would accept the message.

Accepting a message-signature pair at a higher transferability level corresponds to accepting with a higher degree of certainty. By definition, the verification algorithms become more strict as l increases – specifically, for each verifier,

$$\mathsf{Ver}_{(i,l)}(m,\sigma) = \mathsf{True} \quad \Rightarrow \quad \mathsf{Ver}_{(i,l')}(m,\sigma) = \mathsf{True} \quad \text{for all } l' \leq l. \tag{4.1}$$

This must be the case since, if a message is guaranteed to be transferable l times in sequence, it is also guaranteed to be transferable l' times in sequence.

Accepting a message at level l is supposed to provide a guarantee that all other participants would accept the forwarded message at the less strict level $l-1$. In practice, the receiver of the forwarded message may also be able to accept the message at the higher level l, but this is not guaranteed. Thus, the message can be forwarded *at least* l times before level 0 is reached, at which point the message authenticity can be verified but not transferred. The $l = -1$ verification level is necessary for the dispute resolution process discussed below. In short, a level below $l = 0$ is necessary so that, if a message is accepted at level 0 and subsequently a dispute arises, the participants still have a method of collectively deciding the validity of the message.

4.3 Dispute Resolution

For most USS schemes there is no trusted authority. When deciding the validity of a signature, honest participants use the verification algorithm assigned to them by the generation algorithm. In general, since the generation algorithm contains randomness unique to each user, the verification algorithms are also unique to each user. In principle this could be exploited by dishonest coalitions whose aim is to cause two recipients to disagree as to the validity of a message. Even simpler than this, suppose a member of a dishonest coalition decided to reject a forwarded signed message, despite the fact that his true verification algorithm (if it were used) would show the message as valid. Since there is no impartial authority to appeal to, if such

a clash arises how does one decide whether the message is valid?

This question is important in the context of repudiation attempts by Alice. Suppose Bob accepts a message from Alice at a level $l = 0$. This means Bob is convinced the message indeed came from Alice, but that he may not be able to transfer the message to others. Nevertheless, since Bob is convinced, he may accept the message and act on it regardless of the fact that it cannot be transferred[2]. For illustration, consider the scenario that Alice is a software developer who sends Bob a digitally signed software update. Bob has no need to transfer this message, and as long as he is convinced of the sender he will trust and install it. If the package is later found to contain malware and Alice decides to repudiate having sent it, how does one decide who is telling the truth, especially since Bob has no guarantees that he can transfer the message?

This dilemma is solved by incorporating a procedure called *dispute resolution* into the USS scheme. Dispute resolution should be thought of as an expensive last resort akin to taking someone to court. It does not happen in the ordinary run of a protocol, but is present as a safety net to ensure honest participants can prove they acted properly. It is expected that even dishonest participants would be discouraged from pursuing this route, since forcing the expensive dispute resolution process would come with consequences, and the procedure ensures that honest participants prevail so long as they are in the majority.

Definition 4.7 (Dispute resolution)**.** A dispute resolution method DR for a USS scheme $\mathcal{Q}$ is a procedure invoked whenever there is a disagreement on whether a signature σ on a message m is valid. The participant invoking the dispute resolution can be anyone, including the signer P_0. The procedure consists of an algorithm DR that takes as input a message-signature pair (m, σ) and outputs a value $\{\mathsf{Valid}, \mathsf{Invalid}\}$ together with the rules:

1. If $\mathsf{DR}(m, \sigma)$ outputs Valid, then all users must accept (m, σ) as valid.

2. If $\mathsf{DR}(m, \sigma)$ outputs $\mathsf{Invalid}$, then all users must reject (m, σ).

This definition provides a blueprint for the functionality of dispute resolution, but gives no indication of how the DR algorithm could be constructed. Depending on the resources available there are many possibilities of how to construct DR. In the absence of a trusted authority, in this thesis the dispute resolution method used is always *majority vote*. Simply put, all participants use their own verification

[2]In the real world, signatures are most often used in this way – they are used as an authentication scheme, with transferability being a secondary (but still important) functionality.

algorithm at a level $l = -1$ to test if the signature was valid or not. They vote according to the outcome of their test and the majority outcome wins.

Definition 4.8 (Majority Vote)**.** When the validity of a message-signature pair (m, σ) is in dispute, we invoke a majority vote dispute resolution method $\mathsf{MV}(m, \sigma)$, defined by the following rules:

1. $\mathsf{MV}(m, \sigma) = \mathsf{Valid}$ if $\mathsf{Ver}_{(i,-1)}(m, \sigma) = \mathsf{True}$ for more than half of the users.
2. $\mathsf{MV}(m, \sigma) = \mathsf{Invalid}$ otherwise.

The $l = -1$ verification level is reserved for dispute resolution alone. There is nothing particularly special about this level; it is simply there to ensure the existence of a verification level lower than those used for normal runs of the protocol. As we have seen, the lower the verification level, the more lenient the verification algorithm. Therefore, even if a message is considered to be authentic but *not* transferable (i.e. accepted at level 0), by reserving an even lower verification level the protocol still guarantees that an honest participant can prove the message received was authentic.

4.4 Security

As discussed previously in Section 2.3.1, USS schemes must be secure against forging, non-transferability and repudiation. The adversary is not limited to being a single participant, but can instead be any coalition of participants. However, the signer must not be included in the coalition for the notion of forging to make sense, and must be included in the coalition for the notions of repudiation and non-transferability to make sense. Formally, the threats to USS schemes are defined as follows.

Definition 4.9 (Forging)**.** Let $\mathcal{Q}$ be an USS protocol and let $C \subset \mathcal{P}$ be a coalition of malevolent parties that does not include the signer P_0. Suppose that the coalition holds any valid message-signature pair (m, σ) and can use this to output a message-signature pair (m', σ') with $m' \neq m$. We define *Forging* to be the function

$$\mathrm{Forg}_C(\mathcal{Q}, m', \sigma') = \begin{cases} 1 & \text{if } (m', \sigma') \text{ is } i\text{-acceptable for some } P_i \notin C \\ 0 & \text{otherwise.} \end{cases} \tag{4.2}$$

The protocol $\mathcal{Q}$ is ϵ-secure against forging attempts if

$$\sup \left\{ \mathbb{P}(\mathrm{Forg}_C(\mathcal{Q}, m', \sigma') = 1) \right\} \leq \epsilon, \tag{4.3}$$

where the supremum is taken over all possible coalitions and strategies.

The meaning of this definition is that, given access to a single valid message-signature pair, the coalition succeeds in forging (i.e. $\text{Forg}_C(\mathcal{Q}, m', \sigma') = 1$) if they are able to produce a message-signature pair that will be accepted by any honest recipient not part of the coalition. This definition captures the common notion of a forgery in the sense that the coalition, which does not contain the designated sender P_0, is successful if they are able to convince any third party that a message originated with P_0.

Definition 4.10 (Non-Transferability)**.** Let $\mathcal{Q}$ be an USS protocol and $C \subset \mathcal{P}$ a coalition of malevolent participants that includes the signer P_0. Suppose that C outputs a message-signature pair (m, σ) and a verification level l. We define *Non-Transferability* to be the function

$$\text{NonTrans}_C(\mathcal{Q}, m, \sigma, l) = \begin{cases} 1 & \text{if } \mathsf{Ver}_{(i,l)}(m, \sigma) = \mathsf{True} \textit{ for some } P_i \notin C \text{ and} \\ & \mathsf{Ver}_{(j,l')}(m, \sigma) = \mathsf{False} \text{ for some } 0 \leq l' < l \\ & \text{and some } j \neq i, P_j \notin C, \\ 0 & \text{otherwise.} \end{cases} \tag{4.4}$$

The protocol $\mathcal{Q}$ is ϵ-secure against non-transferability if

$$\sup \left\{ \mathbb{P}(\text{NonTrans}_C(\mathcal{Q}, m, \sigma, l) = 1) \right\} \leq \epsilon, \tag{4.5}$$

where the supremum is taken over all possible coalitions and strategies.

The meaning of this definition is that the coalition succeeds in breaking transferability of the scheme (i.e. $\text{NonTrans}_C(\mathcal{Q}, m, \sigma, l) = 1$) if, for any two honest recipients not part of the coalition, they are able to produce a message-signature pair that will be accepted by one of the recipients at some level l, and rejected by the other recipient at the strictly lower level l'. This definition intuitively captures what it means for transferability to be broken in the scheme – the coalition, which includes the sender P_0, is successful if they are able to make an honest recipient believe a message is transferable l times when in fact it is not.

Definition 4.11 (Repudiation)**.** Let $\mathcal{Q}$ be an USS protocol and $C \subset \mathcal{P}$ a coalition of malevolent participants that includes the signer P_0. Suppose that C outputs a

message-signature pair (m, σ). We define *Repudiation* to be the function:

$$\mathrm{Rep}_C(\mathcal{Q}, \mathsf{MV}, m, \sigma) = \begin{cases} 1 & \text{if } (m,\sigma) \text{ is } i\text{-acceptable for some } P_i \notin C \text{ and} \\ & \mathsf{MV}(m,\sigma) = \mathsf{Invalid} \\ 0 & \text{otherwise} \end{cases} \tag{4.6}$$

The protocol $\mathcal{Q}$ is ϵ-secure against repudiation attempts if

$$\sup\left\{\mathbb{P}(\mathrm{Rep}_C(\mathcal{Q}, \mathsf{MV}, m, \sigma) = 1)\right\} \leq \epsilon, \tag{4.7}$$

where the supremum is taken over all possible coalitions and strategies.

The meaning of this definition is that the coalition succeeds in repudiating (i.e. $\mathrm{Rep}_C(\mathcal{Q}, \mathsf{MV}, m, \sigma) = 1$) if they are able to produce a message-signature pair that will be accepted by an honest recipient and yet, if the coalition denies having sent the message, the dispute resolution procedure will rule in favour of the coalition. This definition captures the common notion of repudiation, in which P_0 sends out a message and later tries to deny having sent it.

Lastly, we define what it means for an USS scheme to be secure.

Definition 4.12 (Security of USS schemes)**.** An USS protocol $\mathcal{Q}$ is ϵ-secure if it is ϵ-secure against forging, non-transferability and repudiation attempts.

In other words, the protocol is called ϵ-secure if it is ϵ-secure against all three types of threat. Alternatively, if the probabilities of forging, repudiating and non-transferability all decay exponentially with respect to some security parameter, then we will simply say that the protocol is secure. We will refer to these definitions in later chapters when analysing the security of proposed schemes.

Chapter 5

Considerations for constructing practical USS schemes

5.1 Introduction

At this point in the thesis we will be moving away from discussions of generic USS schemes and instead start to construct new USS schemes. This chapter aims to provide motivation as to why the USS schemes presented in later sections appear the way they do. With the exception of Theorem 5.1 in Section 5.4, the arguments presented in this chapter are not rigorous, but are instead meant to serve as a guide in the search for secure, efficient and practical USS schemes.

Quantum USS scheme template

Quantum USS schemes have been proposed in many different forms throughout the literature, but all *realisable* schemes[1] have followed the same generic template. Namely, in the distribution stage, the sender Alice transmits quantum states to each of the recipients. The recipients perform measurements to gain partial information on the states chosen by Alice. In the messaging stage, Alice's signature is a classical record of the states transmitted which the recipients can verify by checking it against the partial information gained from the states they received.

Throughout this chapter it may be helpful to keep this generic quantum USS template in mind. We do not claim that this is the most general form for quantum USS protocols, but stress that all known realisable quantum USS schemes (at the time of writing this thesis) fit this generic template. Therefore, we have found it useful to consider this template when searching for more efficient USS schemes that

[1] I.e. schemes that can be realised using current technology.

remain realisable with currently available technology.

Security

Roughly speaking, the security of all USS schemes, both classical and quantum, rests on two key features:

1. **Partial information recovery**: recipients do not gain full information on what is sent from/to Alice. This prevents recipients from being able to forge messages. In classical schemes this property is enforced using secret channels to hide selected communications. In quantum schemes this property can also be enforced by having Alice transmit states selected from a non-orthogonal ensemble.

2. **Recipient symmetry**: the information held by each recipient is identically distributed from Alice's perspective. This prevents her from being able to create biased signatures that are more likely to be rejected by one recipient than another, thus safeguarding against both repudiation and non-transferability. In classical schemes this property is often achieved using anonymous channels to hide the identity of the recipients. Quantum schemes have instead used secret channels to perform an exchange process that enforces symmetry (e.g. Protocol 1 from Section 2.4).

5.2 Same-state quantum USS schemes

Many quantum USS schemes, for example Refs. [1, 50–53, 59, 108, 109], involve Alice sending *the same* states to all recipients in the distribution stage. This is analogous to public-key digital signature schemes in which the signature is set up by having the sender broadcast the same information to all possible recipients. Intuitively schemes of this type seem natural because we want all participants to agree on the validity of a signature, and sending the same states to each recipient seems to enforce *recipient symmetry*.

These schemes could also be seen as potentially advantageous because they might allow for quantum USS schemes requiring significantly fewer channels than classical USS schemes, and fewer than those assumed in the standard resource model. Specifically, the same-state quantum USS schemes might seem to require direct quantum channels only from the sender to each recipient (i.e. linear in the number of recipients), rather than pairwise between all participants as given by the standard

resource model. If true, this would be an advantage over classical USS schemes, which all require channels pairwise between all participants, i.e. quadratic in the number of participants[2].

Nevertheless, we shall show in this section that same-state quantum protocols introduce significant problems. In particular, since a t-party coalition of dishonest recipients will have access to t copies of each state, as t increases they are able to recreate each state sent by Alice with high fidelity, and thereby forge messages. This places restrictive limits on the number of colluding adversaries that the protocols are able to handle. Further, as well shall see in the following section, it is difficult to remove the majority of recipient-recipient communication without drastically reducing the efficiency of the protocols.

Recipient symmetry in same-state protocols

Of vital importance to same-state schemes is the notion of *broadcast*. The recipients must be able to check that Alice sends the same states to each participant. If they cannot check this, a dishonest Alice could easily break *recipient symmetry* by sending entirely different states to each participant so that her future signature agrees with what one recipient received in the distribution stage, and yet disagrees with another. So, how can recipients check that they received the same states from Alice in the distribution stage, without revealing too much information to compromise the *partial information recovery* property?

As we shall see, this issue can only be avoided using additional channels to connect the recipients. Assuming that all participants are connected pairwise by classical authenticated channels, the most practical solution would be to implement a sampling procedure in which each participant chooses a selection of the received states, and broadcasts[3] the associated information to other recipients, who can use that information to check that Alice sent out the same states.

However, this method has two problems. First, implementing detectable broadcast requires authenticated classical channels between all participants, partially removing the "fewer channels" advantage hoped for in same-state quantum protocols[4]. On top of this, the communication overhead required to implement detectable broad-

[2]As always, this statement applies only to classical USS schemes that do not use a trusted authority.

[3]Using a detectable broadcast protocol, not an authenticated broadcast protocol (c.f. Section 3.8).

[4]The standard resource model assumes authenticated classical channels *and* insecure quantum channels pairwise between all participants. Therefore the "fewer channels" advantage is not fully lost here.

cast is fairly large and hinders efficiency. The second main problem is that, even if we assume broadcast as an additional resource, for any practical quantum USS protocol there still remains powerful strategies available to Alice which allow her to cheat.

Transferability attacks on same-state protocols

In this subsection we outline an attack available to Alice in same-state protocols when the quantum channels are lossy. We consider a three-party protocol in which: Alice, Bob and Charlie are all connected by authenticated classical channels; Alice-Bob and Alice-Charlie are connected by insecure quantum channels; and there exists a broadcast channel. Again, for the following arguments it is useful to keep the generic quantum USS template in mind.

In any practical setting the quantum channels will be lossy; suppose that states sent over the quantum channels are lost with probability q. If Alice is dishonest, for information-theoretic security we must assume that she can replace these imperfect quantum channels with lossless ones. In the distribution stage, if Alice is supposed to send n states in total, she can instead use the lossless quantum channels to send $n(1-q)$ signals to Bob and $n(1-q)$ signals to Charlie. Alice artificially introduces losses such that the losses of Bob and Charlie do not overlap.

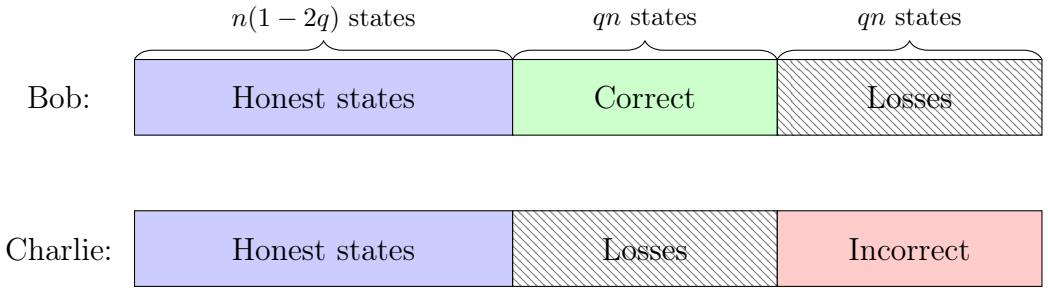


Figure 5.1: Representation of the states sent to Bob and Charlie. The left block represents positions in which both Bob and Charlie received a state. For these states, Alice acts honestly and sends the same thing to both recipients. The middle block represents positions in which Alice sent states to Bob but not to Charlie. The states sent to Bob are chosen by Alice so as to agree with her future signature declaration. The right block represents positions in which Alice sent states to Charlie, but not to Bob. The states sent to Charlie are chosen by Alice so as to disagree with her future signature declaration.

In this way Alice can increase either Bob's or Charlie's error rate (with her future signature) by q. If $q > s_v - s_a$, where s_v and s_a are the message acceptance thresholds at transferability levels 0 and 1 respectively, then Alice can in this way break transferability. In most cases in the literature, $s_v - s_a < 0.1$. This is because most realised quantum schemes have found a channel error rate of above 1%, and the

schemes have been unable to tolerate more than about an 11% channel errors, e.g. Refs. [59, 109–111]. For correctness and security, the parameters s_a and s_v must be chosen to be between these two parameters, i.e. above the expected channel error rate, but below the threshold tolerable channel error for the protocol. Overall, this means that transferability will often be compromised for even small channel loss. Note that Alice's strategy will not be caught by any sampling performed by Bob and Charlie since they agree on all positions where they both received a state. Of course, this strategy can also be applied in the general N-participant scenario.

Preventing loss attacks

A simple method for preventing Alice from performing this type of attack is to have *all* recipients discard *all* signals unless *all* recipients report that they received a signal at a given position. Indeed this strategy has been proposed and implemented in the literature [108, 109]. However, this strategy becomes extremely inefficient and entirely impractical for even moderate numbers of participants – for example, with 10 recipients and a total system loss of just 6 dB for each recipient, less than one in every million states sent would be kept.

The only other known resolution to these loss manipulation strategies is to have recipients secretly exchange a selection of the states (or measurement outcomes) received from Alice. The exact exchange method will depend on the details of the protocol, but the goal is to enforce *recipient symmetry*, regardless of what Alice sends. Of course, this further requires recipients to be connected by quantum channels, either to forward the states directly, or to perform QKD to generate a secret classical channel. In this case, we are back to the standard resource model. Therefore, it seems that same-state protocols cannot simultaneously remain practical *and* use fewer resources than those granted in the standard resource model.

5.3 Exchange-type quantum USS schemes

In this section we consider quantum protocols in which the participants enforce recipient symmetry by exchanging a selection of their measurement outcomes. For simplicity, we again restrict to the three-party scenario. An example of an exchange procedure is described in Step 6 of the distribution stage of Protocol 1 (see Section 2.4). The aim of the exchange procedure is to leave Bob and Charlie with outcomes that have the same expected error rate with whatever signature Alice can later declare. For this to happen, Bob and Charlie must exchange their measurement

outcomes in secret so that Alice cannot selectively introduce errors for one party and not the other.

The end result of the exchange process is that, regardless of what Alice sends, Bob and Charlie have the same expected error rates with Alice's future signature declaration. Therefore, it is natural to ask: is it sensible for protocols to require an honest Alice to send the same states to Bob and Charlie?

Same-state vs different-state protocols

The exchange process ensures that security against repudiation and non-transferability are guaranteed regardless of whether Alice sends the same or different states to each recipient. On the other hand, having Alice send the same states to recipients helps dishonest forgers by weakening the *partial information recovery* property – a dishonest Bob is provided with a perfect copy of the states sent to Charlie. For protocols involving larger numbers of participants the situation is even worse, since dishonest coalitions would have access to many copies of the states sent to honest participants, thereby allowing them to make accurate estimates of exactly what Alice sent.

As such, same-state quantum protocols place highly restrictive limitations on the number of participants allowed in any quantum USS scheme. Instead, it seems more efficient and secure to specify exchange-type protocols in which Alice sends *different* states to Bob and Charlie, similarly to the classical USS scheme P2 [1].

Basis reconciliation

In this subsection we consider Protocol 1 from Section 2.4 and consider how it could be improved. Based on the discussion in the previous subsection, we immediately make the modification that an honest Alice is not required to send the same states to Bob and Charlie. Instead, each state that is sent to Bob or Charlie is chosen independently and uniformly at random from the set $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$.

We further examine whether it is more efficient for recipients to perform the unambiguous state elimination (USE) measurements used in Protocol 1, or whether it is beneficial to include a BB84-style processing stage where the sender and receiver announce their basis choices and only results in matching bases are kept. Based on the discussion in the remainder of this subsection, we will conclude that for a number of reasons it is better to use the latter.

For non-repudiation and transferability, it is the exchange process that ensures security. Without a basis reconciliation step Alice does not know which basis each recipient chose to measure in. Whenever an element of her signature is expressed

in a different basis to the chosen measurement basis, this signature element will not cause an error, regardless of who made the measurement. Effectively, this means Alice's signature contains redundant information that does not provide additional security against these threats. Shorter signatures are desirable because they are appended to the message transmitted, and so carry a communication cost. Performing basis reconciliation allows for a shorter signature lengths and does not compromise security.

For forging, since Alice sends different states to Bob and Charlie, a dishonest Bob's information comes entirely from eavesdropping on the Alice-Charlie channel. Without basis reconciliation, each signature element is taken from the set $S_{USE} = \{0, 1, +, -\}$. When Bob is trying to forge, for each signature element he is trying to choose one of the 3 members of S_{USE} that do not cause a mismatch with Charlie's recorded outcome. Recall that Charlie performs an USE measurement to exclude a single element of S_{USE}, and a mismatch occurs if Bob declares the excluded element. With basis reconciliation, each signature element is taken from the set $S_{BR} = \{0, 1\}$, and Bob is trying to choose one of the two members of S_{BR} that will not cause a mismatch with Charlie's recorded outcome. Therefore, a naïve argument suggests that Bob's task is easier without basis reconciliation, since 3 out of the 4 elements of S_{USE} will not cause a mismatch.

However, this naïve argument may not be correct, since the basis declaration step reveals additional information which Bob may be able to use to help him to forge a message. Nevertheless, as in QKD, it can be shown that so long as the Alice-Charlie quantum channel error rate is reasonably small, the basis declaration does not reveal much information to Bob. For signatures of equal length, the forger's task is indeed harder with basis reconciliation than without. Intuitively, this can be understood as follows. Without basis reconciliation, Bob has more freedom in choosing his forging strategy. Whenever an element of Bob's dishonest signature is specified in a different basis to Charlie's measurement, Charlie will never find a mismatch on that element. In this way, allowing the potential of mismatched bases helps the forger to reduce his overall error rate. Therefore, although the basis declaration reveals some small amount of information to the forger, this is offset by forcing the forger to declare elements in the same basis as measured by the verifier Charlie.

A final benefit to including the basis reconciliation step is that it allows us to leverage existing results in QKD and apply them to quantum USS schemes. As we shall see in Chapter 6, the theoretical tools developed to analyse QKD protocols are powerful, and allow for significant improvements in both the security analysis and

experimental implementations of quantum USS schemes.

5.4 Minimal resource requirements for USS schemes

We end this chapter by considering limitations on two stage (distribution stage and messaging stage) USS schemes performed in which the only resources available to the participants are point-to-point communication channels, i.e. there is no trusted authority or additional resources such as a broadcast channel.

We find that USS schemes in this setting must have an interactive distribution stage[5], and if $O(N)$ participants can be dishonest, then they always require $O(N^2)$ secret keys held pairwise between participants. By interactive, we mean that all recipients must be able to communicate with one another. This is as opposed to non-interactive schemes (e.g. computationally secure public-key digital signature schemes) in which each recipient communicates only with the sender.

Theorem 5.1. *All USS schemes of the above type require participant interaction in the distribution stage. Further, in an N-party scheme allowing $O(N)$ dishonest participants, the number of authenticated channels between protocol participants, and therefore the overall amount of secret shared key required between participants, increases as $O(N^2)$.*

Proof. USS schemes of the type considered here all contain a distribution stage in which information is distributed amongst the protocol participants. Protocol recipients use the distributed information to verify signatures in the messaging stage.

Consider an N-party USS scheme in which two recipients, P_1 and P_2, are entirely separated in the distribution stage and are unable to learn anything about the information held by the other, i.e. they cannot communicate either directly or indirectly using other (trusted or untrusted) protocol participants as intermediaries. In this case, in the messaging stage, the sender is free to choose a signature that will agree with information held by P_1 and disagree with the information held by P_2. Recall that, in the messaging stage, protocol recipients check the signature locally, without any interaction with other recipients.

Since P_1 and P_2 cannot exchange any information in the distribution stage of the protocol, there is no way that they can derive any assurances that the other will accept a signature as valid. Therefore a signature in a scheme where two parties are kept entirely separate cannot be provably transferable. The above argument

[5]This is similar to the case of secret sharing, in which it is impossible to provide unconditional security for all participants using a non-interactive protocol [112].

shows that USS schemes must include at least the potential for either direct or indirect means of communication between *all* pairs of protocol participants in the distribution stage of the protocol.

The protocol will therefore contain at least one communication channel. There must also be at least one authenticated communication channel, since if all communication channels were unauthenticated then a single dishonest adversary could intercept all communications and perform a man-in-the-middle attack to cheat in any desired way.

Each honest participant must be connected via an authenticated communication channel to at least one other honest participant, since otherwise he would have no way of communicating with the other honest participants, since the dishonest participants could perform man-in-the-middle attacks to separate the honest participant from all other honest participants. As above, this would mean that messages cannot be transferable since two honest recipients would be entirely separated.

Therefore, if d_P is the number of dishonest participants allowed by the protocol, each participant must be directly connected via an authenticated channel to at least $d_P + 1$ other participants. To authenticate communication channels with information-theoretic security, it is necessary for the two communicating parties to share a secret key whose size depends on the length of the message being authenticated (see Section 3.6). For $d_P = O(N)$, the number of authenticated direct channels required, and therefore the amount of secret shared key required, must increase quadratically in the number of participants, i.e. the secret shared key requirements scale as $O(N^2)$. □

For example, if the protocol can tolerate up to 1/2 of the participants being dishonest, there must be at least $N^2/4$ direct communication channels (since each channel connects two participants).

5.5 Conclusion

Overall, the arguments presented in this chapter help to guide the construction of new USS schemes in the following chapters. Section 5.2 considered same-state protocols, and argued that in fact the resources assumed in the standard resource model are minimal and necessary for practical quantum USS schemes. Essentially, this followed from the requirement of *recipient symmetry*, which meant that recipients needed the ability to corroborate the information they received from the sender with all other recipients. Without such corroboration, powerful and undetectable

cheating strategies exist for Alice. As such, it seems necessary to have pairwise authenticated classical channels *and* pairwise quantum/secret channels between all participants.

This leads naturally to exchange-type protocols, which are considered in Section 5.3. We re-examined Protocol 1 and argue that, given recipients exchange a selection of their received states, there is no functional benefit to having Alice send the same states out to all recipients. In fact, protocols in which Alice sends the same states to all recipients face prohibitive restrictions on the number of participants allowed in the scheme, since larger collusions can break the *partial information recovery* property and therefore are able to forge. We further found that it is beneficial from an efficiency perspective to include a basis reconciliation step in Protocol 1, leading to a BB84-type measurement process. These considerations lead directly to the protocol described and analysed in Chapter 6.

Lastly, Theorem 5.1 proves that in any USS scheme, all protocol participants must have the ability to interact in the distribution stage. Additionally, it states that the number of authenticated channels required scales quadratically with the number of participants. In Chapter 8 we will see a classical USS scheme which only requires participants to (pairwise) share secret keys that are logarithmic in the size of the message being signed. This requirement is no more expensive than assuming participants are connected pairwise by authenticated classical channels (c.f. Section 3.6). In light of Theorem 5.1, this means that the scheme in Chapter 8 is essentially minimal in terms of resource requirements.

Chapter 6

Secure quantum signatures using insecure quantum channels

6.1 Introduction

Since the original Gottesman-Chuang scheme [49] was proposed in 2001, quantum USS schemes have steadily improved to become simpler, more practical and more efficient. Prior to the work contained in this chapter, Protocol 1 (outlined in Section 2.4) represented the culmination of these advances. However, its security analysis, provided in detail in Ref. [59], was incomplete in two ways. First, the analysis was restricted to collective forging attempts and did not cover coherent attacks. Second, the analysis assumed "tamper-proof" quantum channels that do not allow eavesdropping or modification of the transmitted states. This strong and generally undesirable assumption meant that a potential forger (Bob) only had access to his own copy of the signature states (sent by Alice). In reality an adversarial Bob would be able to gain extra information on Alice's signature through eavesdropping on the signature states sent from Alice to Charlie.

We use this chapter to introduce a new quantum USS scheme – called the AWKA scheme – derived from Protocol 1, but containing three key modifications based on the considerations in Chapter 5.

1. Alice does not send the same states to Bob and Charlie; instead, she sends different states. This has the advantage of making the protocol much more efficient by limiting a dishonest coalition's forging potential. It also enables us to make our second modification, stated next.

2. Although Alice is still the one sending and signing messages in the messaging stage, in the distribution stage it is Bob and Charlie who send the states and

Alice who receives them. In a sense, it is Bob and Charlie who create Alice's signature. This has the practical benefit of making the receiver loss/detector efficiency the same for each participant, since it is always Alice receiving the states. It also removes Alice's ability to send correlated states to Bob and Charlie in the distribution stage.

3. Rather than the USE measurements performed in Protocol 1, we include a BB84-style basis reconciliation step which allows us to both decrease the signature length and to use existing results taken from QKD.

A direct result of the third modification is that we are able to use decoy-state techniques (see Section 3.5) to make the scheme fully realisable with current technology. Further, we are able to prove the security of the AWKA scheme against *all* types of attack, while also removing *all* trust assumptions on the quantum channels. This resolves both issues in the security analysis of Protocol 1, and closes the theory-experiment gap discussed in Section 2.4.3. Our analysis also highlights an interesting theoretical result; namely, in the AWKA protocol the error threshold for the Alice-Bob and Alice-Charlie quantum channels is less strict than that required for distilling a secret key using QKD.

Lastly, the protocol presented in this chapter can be performed using exactly the same equipment as required by QKD. This is of practical benefit for both signatures and QKD. On the signatures side it allows us to make use of the already mature QKD technologies to easily implement our scheme. On the QKD side, signature schemes provide an additional functionality to complement existing QKD networks. The work presented in this chapter has been published in Ref. [110] with minor modifications.

6.2 The AWKA USS scheme

In this chapter we describe the AWKA protocol for three parties, a sender, Alice, and two receivers Bob and Charlie. Generalisation to more parties is possible, but special care should be taken to address colluding adversaries. As before, in the three-party scenario, at most one party can be dishonest, since two colluding dishonest parties can trivially cheat on the third party. We employ a majority vote dispute resolution process, meaning transferability and non-repudiation become identical in this three-party setting.

Scheme outline

We assume that between Alice and Bob, and between Alice and Charlie there exists authenticated classical channels as well as untrusted, imperfect quantum channels. In addition, Bob and Charlie share a QKD link which can be used to transmit classical messages in full secrecy. The protocol makes use of a key-generating protocol (KGP) performed in pairs separately by Alice-Bob and Alice-Charlie. The KGP is essentially a restricted form of QKD, used by the sender and receiver to generate a raw key. They do not proceed to perform the post-processing steps of error correction and privacy amplification. The KGP uses the untrusted quantum channels, and generates two correlated bit strings, one for the sender and one for the receiver. When the channel noise level is below a prescribed threshold, we show that the Hamming distance between the receiver's string and the sender's string is smaller than the Hamming distance between any string an eavesdropper could produce and the sender's string. The KGP is discussed in Section 6.3, after we present the signature protocol itself.

6.2.1 The protocol

As is always is the case for USS schemes, the AWKA protocol has two parts, a distribution stage, where the scheme is set up, and a messaging stage, where messages are signed and sent. The distribution stage involves both classical and quantum communication, whereas all communication in the messaging stage is classical. In this chapter we show how to sign a 1-bit message. Longer messages can be signed by suitably iterating the 1-bit protocol, as in [113].

Distribution stage

1. For each possible future message, $m = 0$ or $m = 1$, Alice independently performs the KGP, twice with Bob and twice with Charlie, to generate four different length L keys, A_B^m and A_C^m, for $m \in \{0, 1\}$ and where the subscript denotes the participant with whom she performed the KGP. Bob holds the two length L strings B^m and Charlie holds the two length L strings C^m.

2. For each value of m, Bob and Charlie each separately and randomly split their keys into two equal parts to obtain the sets B_1^m, B_2^m, C_1^m and C_2^m. Using a secret classical channel, they each forward the set indexed "2" to the other participant so that Bob holds B_1^m and C_2^m, while Charlie holds C_1^m and B_2^m.

For each possible future message m, Alice's signature will be the $2L$ length string $\text{Sig}_m = (A_B^m, A_C^m)$. As we shall show in Section 6.3, except with negligible probability A_B^m contains fewer mismatches with B^m than does any string an eavesdropper, Eve (who may be Charlie) can produce. The same applies to A_C^m and C^m. Essentially, this is what will protect against forging; Alice knows the pair (B^m, C^m) better than anyone else.

Bob and Charlie will check Alice's signature by matching it against the two sets they hold (e.g. Bob uses B_1^m and C_2^m). In Section 6.4 we show that, since Alice does not know how Bob and Charlie split B^m and C^m, the exchange process means that Alice has no information on whether each element in her signature will ultimately be checked by Bob or Charlie. Essentially, this is what will protect against repudiation/non-transferability.

Messaging stage

1. To send a signed 1-bit message m, Alice sends (m, Sig_m) to the desired recipient (say Bob).

2. Bob checks (m, Sig_m) separately against B_1^m and C_2^m[1]. If the signature element is $a \in \{0, 1\}$ and Bob's corresponding stored element is $b \in \{0, 1\}$, a mismatch occurs if $a \neq b$. Bob records the number of mismatches he finds for each of the two sets he checks. If there are fewer than $s_a(L/2)$ mismatches in both sets (where $s_a < 1/2$ is a small threshold determined by the parameters and the desired security level of the protocol) then Bob accepts the message.

3. To forward the message to Charlie, Bob forwards the pair (m, Sig_m) that he received from Alice.

4. Charlie tests for mismatches in the same way (using C_1^m and B_2^m), but in order to protect against repudiation by Alice he uses a different threshold, s_v. Charlie accepts the forwarded message if the number of mismatches in both sets is below $s_v(L/2)$ where $0 < s_a < s_v < 1/2$.

That the recipients must use different thresholds or acceptance criteria for messages received directly from the sender and for forwarded messages is a necessary

[1]Note that Bob could use B_2^m to further check the validity of the signature (and similarly for Charlie). This has some subtle security advantages even in the three-party case, and could protect against forms of collusion in the multi-party case. However, we do not specify this here, since it is not necessary under our three-party security assumptions, and it simplifies our security analysis to consider only the symmetrized keys. Note also that, of course, a dishonest Bob must be assumed to retain full knowledge of B_2^m.

feature of all USS schemes (see Refs. [33, 34]).

6.3 The key generation protocol

We now describe how two parties, Alice and Bob, perform the KGP. Essentially, Alice and Bob perform the quantum part of QKD to generate raw keys, but they do not proceed to error correction or privacy amplification. This means that Alice and Bob will generate different (but correlated) strings that are not entirely secret. These keys are the A_B^m and B^m strings described above. Although the KGP builds on QKD, the security analysis for the KGP does not follow directly from the security of the QKD protocol. This is because the goal of an adversary in the signature protocol is different from that of an eavesdropper in QKD. For the signature protocol, what matters is the number of mismatches with a recipient's key; for QKD, what matters is the information an eavesdropper can hold about the key. Note also that for signatures the eavesdropper acts from within the protocol, and, for example, Eve could be Charlie. This means the eavesdropper has access to additional "side information" over and above that held by an eavesdropper in QKD.

Nevertheless, starting from the bound on an eavesdropper's min-entropy in QKD, we show how to bound the number of mismatches (with B^m) a forger in our signature protocol can achieve. Let $d(.,.)$ be the Hamming distance between two bit strings. We say that the KGP is ϵ-secure if

$$\sup\left\{\mathbb{P}\Big(d(A_B^m, B^m) \geq d(E_{\text{guess}}, B^m)\Big)\right\} \leq \epsilon, \tag{6.1}$$

where the supremum is taken over all strategies for Eve allowed by quantum mechanics and the probability is taken over all the randomness in the protocol. The meaning of this definition is that, except with probability ϵ, the eavesdropper produces a string that contains more mismatches with B^m than does Alice's string.

In this section we prove that the KGP is secure. The security of the overall signature protocol (in which the KGP is used as a subprotocol) will be proven below in Section 6.4.

6.3.1 Implementing the KGP

In what follows, the underlying QKD protocol upon which the KGP is built will be the prepare-and-measure decoy-state BB84 protocol using weak coherent pulses, described in [114]. Specifically, we assume that Bob has a phase-randomised source

of coherent states. The intensity of each light pulse is chosen by Bob to be either u_1, u_2, or u_3, where $u_1 > u_2 > u_3$. The intensities are chosen with probabilities p_{u_1}, p_{u_2}, and p_{u_3}. All intensity levels are used for key generation. To encode information, Bob randomly selects one of four possible polarisation states – $|0\rangle, |1\rangle$ (Z basis) and $|+\rangle := 1/\sqrt{2}(|0\rangle + |1\rangle), |-\rangle := 1/\sqrt{2}(|0\rangle - |1\rangle)$ (X basis). The X and Z bases are chosen with probabilities $p_X \geq 1/2$ and $p_Z = 1 - p_Z \leq 1/2$ respectively. The asymmetric probabilities for the two bases can be used to increase the efficiency of the protocol [115]. Intensities and states are chosen independently by Bob to avoid correlations between intensity and information encoding. Alice also independently chooses the X and Z measurement bases with probabilities p_X and p_Z respectively. Notice that it is Bob who prepares the states and sends them along the quantum channel to Alice. This role reversal may not be necessary, but simplifies the security analysis in two ways:

1. A dishonest Alice cannot send correlated states to Bob and Charlie.

2. Receiver loss and detector efficiency will be the same for both the Alice-Bob KGP and the Alice-Charlie KGP, since both use Alice as the receiver.

For each state sent by Bob, Alice obtains one of four possible outcomes $\{0, 1, \emptyset, d\}$, where 0 and 1 are the bit values, $\emptyset$ represents no detection and d is a double click event. In the case of double clicks, there is an additional post-processing stage in which Alice randomly assigns the double click to a single bit value, in line with the squashing model [116]. Alice and Bob then announce their basis and intensity choices over an authenticated classical channel. If states are transmitted and then measured in different bases, or if there is no detection, they are discarded (sifting). The protocol is continued until a sufficient number of measurement outcomes have been obtained for each basis and intensity choice.

A raw key is generated by choosing a random sample of size $L+k$ of the X basis counts. The bit string generated by Bob is split into three parts $(V_B, X_{B,\text{keep}}, X_{B,\text{forward}})$ of length k, $L/2$ and $L/2$ respectively. Alice holds the corresponding strings (V_A, X_A), where V_A has length k and X_A has length L. Note that she does not know which bits Bob chooses to forward and which he chooses to keep, but she does know the index positions of the counts in V_B. As in QKD, the V strings are used to perform *parameter estimation* to estimate the correlation between Alice's and Bob's strings generated from X basis measurements, after which they are discarded. The two strings, $X_{B,\text{keep}}$ and $X_{B,\text{forward}}$, refer to Bob's keys, B_1^m and B_2^m, respectively[2]. To

[2]We believe that the duplication of notation is justified by the additional clarity it provides.

ease notation we set $n := L/2$.

Alice and Bob also randomly select a sample of Z basis counts, which we denote Z_A and Z_B, respectively. These strings are used to quantify the level of eavesdropping by Eve. Essentially, Eve's smooth min-entropy on $X_{B,\text{keep}}$ can be quantified using the entropic uncertainty relations described in Section 3.4.4 together with the level of correlation between Z_A and Z_B.

It should be stressed that, contrary to all QKD protocols, in USS schemes it cannot be assumed that Alice and Bob are honest. However, as will be explained below, neither can gain from dishonesty during the KGP.

6.3.2 Security of the KGP

In what follows we consider a finite number of states being sent and measured. Eve is allowed to perform the most general attack permissible by quantum mechanics – a so-called "coherent" attack. This means that Eve can perform any operation allowed by quantum mechanics on any/all states sent over the quantum channel, as well as an arbitrary ancilla system she prepares. Eve is also able to hold systems in quantum memory and perform general measurements at any point during or after the protocol. In this way she is free to take full advantage of all communications, both classical and quantum, sent between Alice and Bob. The classical random variables V, Θ^n and $X_{B,\text{forward}}$ represent the information gained by Eve from parameter estimation, basis declarations in the sifting step and, if Eve is Charlie, the forwarding of $X_{B,\text{forward}}$ by Bob, respectively. Our strategy is to find Eve's information in terms of her smooth min-entropy, and use that to bound the probability that she can make a signature declaration containing fewer than a specified number of mismatches with Bob's key.

Eve's smooth min-entropy

Eve's conditional smooth min-entropy on Bob's key $X_{B,\text{keep}}$ can be derived using existing results in QKD, with the only difference being that here Bob gives the extra information $X_{B,\text{forward}}$ to Eve. However, since Bob does not subsequently use this part of the key, this can be treated in the same manner as the V string sacrificed for parameter estimation [117]. For ease of notation, we will simply write X instead

In quantum information it is common for letters near the start of the alphabet (A, B, C, etc) to refer to quantum systems, whereas letters near the end of the alphabet (X, Y, Z, etc) refer to classical random variables. For this reason, during the KGP subprotocol, to align with standard QKD notation we denote Bob's keys using X, since they are classical bit strings generated from X basis measurements. Nevertheless, when discussing the full signature protocol it is clearer to denote Bob's keys using the B label to denote Bob's identity.

of $X_{B,\text{keep}}$.

We gather all of Eve's information into one quantum system living in the Hilbert space $\mathcal{H}_E$. This comprises the space containing Eve's ancilla quantum systems following her coherent attack, $\mathcal{H}_{E'}$, as well as the spaces containing the classical information V, Θ^n, and $X_{B,\text{forward}}$, which we assume are available to Eve. As in Appendix B of [114], the min-entropy is then

$$H_{\min}^{\epsilon}(X|E) \gtrapprox s_{X,0}^{-} + s_{X,1}^{-}\big[1 - h(\phi_{X,1}^{+})\big], \tag{6.2}$$

where the inequality holds up to a small additive term proportional to $\log(1/\epsilon)$. Here $s_{X,0}^{-}$ and $s_{X,1}^{-}$ are estimates of the number of X basis counts which come from 0 and 1-photon pulses respectively, and which make up the entries in the string X. $\phi_{X,1}^{+}$ is the phase error rate in X basis measurements coming from single-photon pulses. The superscripts $+$ and $-$ are upper and lower bounds representing worst-case scenario estimates consistent with parameter estimation performed on a finite sample (see Appendix A.1), and h is the binary entropy.

Guessing bounds

Given Eve's conditional smooth min-entropy, the following theorem places bounds on Eve's ability to guess X to within a certain Hamming distance.

Theorem 6.1. *Suppose that Bob and Eve share the state ρ_{XE} where, as above, X is an n-bit string held by Bob and E is a quantum system representing all information held by Eve. Then, for any strategy, Eve's probability of making at most r mistakes when guessing X can be bounded as*[3]

$$p_r \leq \sum_{k=0}^{r} \binom{n}{k} 2^{-H_{\min}^{\epsilon}(X|E)_{\rho}} + \epsilon. \tag{6.3}$$

To prove this theorem, we use the following two lemmas which are proved in

[3]Note that, compared to Ref. [110], this thesis makes a subtle change to the notion of security, to one which we now believe makes more sense. Both here and in Ref. [110], Eve succeeds if she is able to make at most r mistakes when guessing X. As per the proof of Lemma 6.3, Eve uses the value of a random variable F to guess X. Although F is a random variable, its distribution function P_F depends on Eve's strategy. In this thesis, we have defined Eve's success probability, p_r, to be her probability of making at most r mistakes when guessing X, *averaged* over P_F. In Ref. [110], a stricter notion of p_r was used – namely, instead of averaging over P_F, it was shown that Eve could not succeed for any F outcome, except with some small probability. Since, given P_F, Eve cannot further control the value taken by F, we believe the averaged definition used throughout this chapter and the next makes more sense. The ideas and essence of the security proofs remain the same under either definition, but the averaged definition used in this thesis allows for a clearer and simpler statement of our results.

Appendix A.2.

Lemma 6.2. *Let τ_{XF} be a classical state, i.e.*

$$\tau_{XF} = \sum_{x,f} P_{XF}(x,f) \left|x\right\rangle \left\langle x\right| \otimes \left|f\right\rangle \left\langle f\right| \tag{6.4}$$

for some orthonormal bases $\{|x\rangle\}_x$ and $\{|f\rangle\}_f$. Let $B^\epsilon(\tau_{XF})$ denote the set of all sub-normalised density matrices ϵ-close to τ_{XF} in terms of the generalised purified distance. Then

$$H^\epsilon_{min}(X|F)_\tau = H_{min}(X|F)_{\overline{\tau}} \tag{6.5}$$

for some classical $\overline{\tau}_{XF} \in B^\epsilon(\tau_{XF})$.

Lemma 6.3. *Suppose Bob and Eve share the classical state η_{XF} defined by the probability distribution Q_{XF}, with Bob holding X and Eve holding F. Let q_r be Eve's probability of guessing X making fewer than r errors, given that X and F are distributed according to Q_{XF}. Then q_r can be bounded as*

$$q_r \leq \sum_{k=0}^{r} \binom{n}{k} 2^{-H_{min}(X|F)_\eta}. \tag{6.6}$$

Notation 6.4. For the sake of readability, we introduce the notation

$$b^r_n := \sum_{k=0}^{r} \binom{n}{k}. \tag{6.7}$$

Proof of Theorem 6.1. Bob and Eve share the state ρ_{XE} and Eve aims to use this to guess X while making fewer than r errors. Since Eve must output a classical string, she performs some optimal CPTP mapping $\mathcal{N}^{E\to F}$ to transform system E into a classical random variable, F, which dictates her guess for X[4]. Her strategy maps

$$\rho_{XE} \to \tau_{XF} := \sum_{x,f} P_{XF}(x,f) \left|x\right\rangle \left\langle x\right| \otimes \left|f\right\rangle \left\langle f\right|, \tag{6.8}$$

where P_{XF} is a probability distribution. Although τ_{XF} (and hence P_{XF}) are unknown, Lemma 6.2 states that

$$H^\epsilon_{\min}(X|F)_\tau = H_{\min}(X|F)_{\overline{\tau}}, \tag{6.9}$$

for some classical $\overline{\tau}_{XF} \in B^\epsilon(\tau_{XF})$ defined by the (possibly sub-normalised) proba-

[4]For example, F could simply represent Eve's guess for X. More generally though, F could just be a classical string that, following some processing, leads to Eve's guess for X.

bility distribution $\overline{P}_{XF}$. Suppose that $\mathrm{Tr}(\overline{\tau}_{XF}) = 1 - \delta$. Then if X and F were distributed according to the probability distribution $Q_{XF} := \frac{1}{1-\delta}\overline{P}_{XF}$, applying Lemma 6.3 gives

$$q_r \leq \frac{1}{1-\delta} b_n^r 2^{-H_{\min}(X|F)_{\overline{\tau}}}, \tag{6.10}$$

where q_r is Eve's probability of making up to r errors under Q_{XF}. In fact, X and F are distributed according to P_{XF}, so we would like to use Eq. (6.10) to bound p_r. The purified distance upper-bounds the trace distance, and the trace distance characterises the distinguishability of probability distributions. Since P_{XF} is ϵ-close to $(1-\delta)Q_{XF}$ in terms of the purified distance,

$$p_r \leq (1-\delta)q_r + \epsilon. \tag{6.11}$$

This means that

$$p_r \leq b_n^r 2^{-H_{\min}(X|F)_{\overline{\tau}}} + \epsilon. \tag{6.12}$$

The above expression is still not particularly enlightening, since $\overline{\tau}$ is unknown in general. Nevertheless, the data processing inequality (Section 3.4.4) and Lemma 6.2 give

$$H_{\min}^{\epsilon}(X|E)_{\rho} \leq H_{\min}^{\epsilon}(X|F)_{\tau} = H_{\min}(X|F)_{\overline{\tau}}. \tag{6.13}$$

Putting it all together, we can bound p_r as

$$p_r \leq b_n^r 2^{-H_{\min}^{\epsilon}(X|E)_{\rho}} + \epsilon. \tag{6.14}$$

□

6.3.3 Application to signatures

In the preceding section we were able to bound Eve's probability of guessing X (to within r mistakes) in terms of her conditional smooth min-entropy. For large values of n we can simplify this bound using the approximation $b_n^r \approx 2^{nh(r/n)}$. Combining Theorem 6.1 with the expression for the min-entropy given in Eq. (6.2), and defining $\gamma := r/n$ to be the mismatch rate, we find[5]

$$p_r \leq 2^{-n\left\{c_{X,0}^{-} + c_{X,1}^{-}[1-h(\phi_{X,1}^{+})] - h(\gamma)\right\}} + \epsilon, \tag{6.15}$$

[5]The equation above should technically have an approximation sign since we have used the approximate bound on the min-entropy from Eq. (6.2). It can be made exact by including the terms proportional to $\log(1/\epsilon)$ in the min-entropy. However, for simplicity and since they do not affect our results, we have neglected these small terms.

where $c^-_{X,i} := s^-_{X,i}/n$ is a lower bound on the expected number of counts per pulse that arise from an X basis signal state containing i photons. The condition

$$c^-_{X,0} + c^-_{X,1}[1 - h(\phi^+_{X,1})] - h(\gamma) > 0 \tag{6.16}$$

determines whether or not Eve is able to make errors at a rate smaller than γ with non-negligible probability. If the condition holds, n can be increased to make Eve's probability of making errors at a rate smaller than γ arbitrarily small (and decay exponentially fast). We will see in the following section that this means that Eve's probability of successfully forging a message can also be made arbitrarily small. We define p^*_E by the equation

$$c^-_{X,0} + c^-_{X,1}[1 - h(\phi^+_{X,1})] - h(p^*_E) = 0, \tag{6.17}$$

i.e. p^*_E is the error rate such that the left hand side of Eq. (6.16) equals zero. The meaning of this is that p^*_E is the minimum fraction of errors that Eve will be able to make when trying to guess $X_{B,\text{keep}}$.

Suppose the error rate on X basis measurements between Alice and Bob is upper bounded as e^+_X (recall that this bound is found from parameter estimation on V_A and V_B). As long as $p^*_E > e^+_X$, there exists a choice of parameters and signature length which make the protocol secure to any security level (see Section 6.4). Equivalently, quantum USS are possible as long as

$$c^-_{X,0} + c^-_{X,1}[1 - h(\phi^+_{X,1})] - h(e^+_X) > 0. \tag{6.18}$$

6.4 AWKA protocol security analysis

We will now prove the robustness and security of the main signature protocol, the AWKA protocol, described in Section 6.2. For robustness, we prove that the probability of the signature being rejected when all participants are honest is negligibly small. For security, we prove that the probability of an adversary being able to forge or repudiate (as per Definitions 4.9 and 4.11) is negligibly small. Recall that for this three-party protocol using the majority vote dispute resolution process, security against repudiation and non-transferability are equivalent.

In what follows, we assume that Bob and Charlie have each independently performed the KGP twice (once for each future message) with Alice to generate the strings $(V_{B,m}, Z_{B,m}, X_{B,m,\text{keep}}, X_{B,m,\text{forward}})$ for Bob and

$(V_{C,m}, Z_{C,m}, X_{C,m,\text{keep}}, X_{C,m,\text{forward}})$ for Charlie, where $m \in \{0,1\}$ denotes the message. Translated to the USS notation as per Section 6.2:

- for Bob, $X_{B,m,\text{keep}} = B_1^m$ and $X_{B,m,\text{forward}} = B_2^m$. The corresponding L-bit string generated by the KGP for Alice is A_B^m.
- for Charlie, $X_{C,m,\text{keep}} = C_1^m$ and $X_{C,m,\text{forward}} = C_2^m$. The corresponding L-bit string generated by the KGP for Alice is A_C^m.

6.4.1 Robustness

Suppose that all participants are honest. Bob rejects a signed message if either B_1^m or C_2^m has a mismatch rate higher than s_a with Alice's signature, (A_B^m, A_C^m). During parameter estimation performed on the strings $V_{A,m}$ and $V_{B,m}$ (both are length k strings) Alice and Bob observe the error rate in the Alice-Bob channel. Their aim is to use these observations to bound the true channel error rate. For this, Serfling's inequality is helpful.

Theorem 6.5 (Serfling's Inequality [118])**.** *Let $X_1, \ldots, X_n$ be a list of random variables taking values in $\{0,1\}$ and let $X_{i_1}, \ldots, X_{i_k}$ be a sample of k of those random variables, chosen without replacement. Further, define*

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i \quad \textit{and} \quad S_k = \sum_{j=1}^{k} X_{i_j}. \tag{6.19}$$

Then for any $\delta > 0$,

$$\mathbb{P}(\mu - \frac{1}{k}S_k \geq \delta) \leq \exp\left[-\frac{2k\delta^2}{1 - \frac{k-1}{n}}\right]. \tag{6.20}$$

Suppose the error rate observed by Alice and Bob during parameter estimation is $\tilde{e}_{X,B}$. Serfling's inequality upper-bounds the true error rate, $e_{X,B}$, in the Alice-Bob channel as

$$e_{X,B} \leq \tilde{e}_{X,B} + \delta := e_{X,B}^{+}, \tag{6.21}$$

where

$$\delta := \sqrt{\frac{\ln(1/\epsilon_{PE})}{2k}\left(1 - \frac{k-1}{n}\right)}. \tag{6.22}$$

The bound holds except with probability ϵ_{PE}. Similarly, we can upper bound the error rate in the Alice-Charlie channel by $e_{X,C}^{+}$. Based on these error rates derived

in the distribution stage of the protocol, the participants set $e_X^+ := \max\{e_{X,B}^+, e_{X,C}^+\}$ and choose s_a such that $s_a > e_X^+$.

Finally, given that the true error rates in both the Alice-Bob and Alice-Charlie channels are less than e_X^+ (except with probability ϵ_{PE}), we can use Hoeffding's inequality to bound the probability of Bob finding an error rate higher than s_a when checking either B_1^m or C_2^m against Alice's signature. The result is

$$\mathbb{P}(\text{Honest Failure}) \leq 2\exp\left[-(s_a - e_X^+)^2 L\right] + 2\epsilon_{PE}, \tag{6.23}$$

where the factors of 2 arise since the abort can be due to either B_1^m or C_2^m.

6.4.2 Security against forging

It is easier for either Bob or Charlie to forge than for any other external party, and we will therefore consider forging by an internal party. Forging is defined in Definition 4.9 in Chapter 4. For the three party scenario considered, the coalition can have at most one member and it cannot be Alice. Without loss of generality, suppose the forger is Bob. In order to forge a message, Bob must give a declaration (m, Sig_m) to Charlie that has fewer than $s_v n$ $(= s_v L/2)$ mismatches with both C_1^m and B_2^m[6]. Since Bob knows B_2^m, matching that part is trivial and we therefore only consider C_1^m. If parameter estimation is successful in the KGP, then the worst-case (maximum) rate at which Alice's signature would make errors with Charlie's key is known – call it e_X^+. From Eq. (6.17), we also know the minimum rate at which a dishonest Bob will make errors with Charlie's key – call it p_E^*.

Assuming $e_X^+ < p_E^*$ (if not, the protocol is aborted), the particpants choose s_v such that $e_X^+ < s_v < p_E^*$. In this case, Charlie will likely accept a legitimate signature originating from Alice, since the upper bound on their error rate, e_X^+, is less than the threshold s_v. On the other hand, Charlie will likely reject any dishonest signature declaration by Bob, since the probability of Bob finding a signature with an error rate smaller than s_v is restricted by Theorem 6.1 as

$$\begin{aligned}
&\mathbb{P}(\text{Bob (Eve) makes fewer than } s_v n \text{ errors}) = p_{s_v n} \\
&\quad \leq 2^{-\left\{H_{\min}^{\epsilon}(X|E)_\rho - nh(s_v)\right\}} + \epsilon \\
&\quad \leq 2^{-n\left\{c_{X,0}^- + c_{X,1}^-[1-h(\phi_{X,1}^+)] - h(s_v)\right\}} + \epsilon.
\end{aligned} \tag{6.24}$$

The calculation of quantities in the above equation involves a number of parameter

[6]Since Charlie creates C^0 and C^1 independently, using arguments very similar to above it can easily be shown that knowledge of $(m', \text{Sig}_{m'})$ cannot help Bob to forge for $m' \neq m$.

estimation processes due to the finite samples taken. Suppose that if any parameter estimation procedure fails (so, for example, if e_X^+ is not a good upper bound), then Bob is able to successfully forge with certainty. In this conservative setting, Bob's probability of successfully forging is bounded as

$$\begin{aligned}\mathbb{P}(\text{Forge}) &\leq 2^{-\left\{H_{\min}^{\epsilon}(X|E)_{\rho}-nh(s_v)\right\}}+\epsilon+\tilde{\epsilon}_{PE}\\ &\leq 2^{-n\left\{c_{X,0}^{-}+c_{X,1}^{-}[1-h(\phi_{X,1}^{+})]-h(s_v)\right\}}+\epsilon+\tilde{\epsilon}_{PE},\end{aligned} \tag{6.25}$$

where $\tilde{\epsilon}_{PE}$ is set as the probability of the upper/lower bounds failing on *any* of the estimated quantities e_X, $s_{X,0}, s_{X,1}$ or $\phi_{X,1}$ (see Appendix A.1). This equation is valid for any choice of $\epsilon, \tilde{\epsilon}_{PE} > 0$ and can be made arbitrarily small by increasing the signature length.

Note that security against forging from Bob derives entirely from the Alice-Charlie KGP, in which Bob is already assumed to be an adversary. Specifically, all parameters which go into the above security analysis can be derived in the distribution stage by Alice and Charlie alone, and cannot be influenced by Bob in any way that will help him to forge.

To make the protocol secure also against attempts to forge from Charlie, exactly the same arguments as above apply except with the roles of Bob and Charlie switched. The overall protocol would find two pairs of e_X^+ and p_E^*, one pair from the Alice-Bob KGP and one from the Alice-Charlie KGP. It would then take the worst case estimates, i.e. set $e_X^+ := \max\{e_{X,B}^+, e_{X,C}^+\}$ and $p_E^* := \min\{p_{E,B}^*, p_{E,C}^*\}$.

6.4.3 Security against repudiation

Repudiation is defined in Definition 4.11 in Chapter 4. Since the dishonest coalition can have at most one member and must include the signer, Alice must be the one trying to repudiate. Without loss of generality, suppose she first sends the message to Bob. In this case, she aims to send a declaration (m, Sig_m) which Bob will accept and, when forwarded, Charlie will reject. To do this, Bob must accept both B_1^m and C_2^m at threshold s_a, and Charlie must reject at least one of either C_1^m or B_2^m at threshold s_v, where $s_v > s_a$.

Intuitively, security against repudiation follows because of the symmetrisation performed by Bob and Charlie using the secret classical channel, and the gap between s_a and s_v. Even if Alice knows and can control the error rates between A_B^m and B^m, and between A_C^m and C^m, she cannot control whether the introduced errors end up with Bob or Charlie following symmetrisation. Accordingly, after symmetrisation,

the expected error rate for Bob with Alice's signature must be the same as the expected error rate for Charlie with Alice's signature. To repudiate, due the the gap between s_a and s_v, one recipient must find significantly more errors than the other.

We give Alice full power and assume that in the messaging stage she is able to fully control the number of mismatches her signature declaration contains with B^m and C^m – call the mismatch rates e_B and e_C respectively. In the symmetrisation process, Bob creates B_2^m by randomly selecting half of the elements in B^m. He sends this set to Charlie in secret. Charlie does similar as per Step 2 of the distribution stage. We aim to show that any choice of e_C and e_B from Alice leads to an exponentially decaying probability of repudiation.

Case 1. Suppose that Alice chooses $e_C > s_a$. In this case, Bob is receiving (without replacement) $L/2$ elements from the set C^m, which contains exactly $e_C L$ mismatches with Alice's future signature declaration (since a dishonest Alice can perfectly control the mismatch rate). The number of mismatches Bob receives in C_2^m therefore follows a hypergeometric distribution $H(L, e_C L, L/2)$, with expected value $e_C L/2$. In order to accept the message, Bob must find fewer than $s_a L/2$ errors. Using tail bounds due to Chvátal [62] we can bound the probability that C_2^m contains fewer than $s_a L/2$ mismatches as

$$\mathbb{P}(C_2^m \text{ contains fewer than } s_a L/2 \text{ mismatches}) \leq \exp[-(e_C - s_a)^2 L]. \tag{6.26}$$

To repudiate, Alice must make Bob accept the message, which means that Bob must accept both B_1^m and C_2^m. Accordingly, the probability of Bob accepting the message is less than or equal to the probability of Bob accepting C_2^m, given by Eq. (6.26), which decays exponentially.

Case 2. Suppose that Alice chooses $e_C \leq s_a$. In this case, if $e_B > s_a$, the same argument as in Case 1 above shows that it is highly likely that Bob will reject the message, so we consider only the case where we also have $e_B \leq s_a$.

Consider the set B^m. We can use the same arguments as above to bound the probability of B_2^m containing more than $s_v L/2$ mismatches as

$$\mathbb{P}(B_2^m \text{ contains more than } s_v L/2 \text{ mismatches}) \leq \exp[-(s_v - e_B)^2 L]. \tag{6.27}$$

Charlie rejects the signature if he finds more than $s_v L/2$ mismatches in either C_1^m or B_2^m. The probability of this happening is at most the sum of the probability of

Charlie finding more than $s_v L/2$ mismatches in C_1^m and the probability of Charlie finding more than $s_v L/2$ mismatches in B_2^m. For $e_B, e_C \leq s_a$, we have

$$\mathbb{P}(\text{Charlie rejects signature}) \leq 2\exp[-(s_v - s_a)^2 L]. \tag{6.28}$$

So again, the probability of Alice successfully repudiating decreases exponentially in the size of the signature. Therefore, we can see that there is no strategy available to Alice that leads to a non-negligle success probability, meaning the protocol is secure against repudiation (and non-transferability) attempts.

In fact, Alice's optimal strategy is to choose the middle ground and set $e_B = e_C = \frac{1}{2}(s_v + s_a)$. In this case

$$\mathbb{P}(\text{Repudiation}) \leq 2\exp\left[-\frac{1}{4}(s_v - s_a)^2 L\right]. \tag{6.29}$$

Note that security against repudiation derives entirely from the symmetrisation performed by Bob and Charlie, in which Alice plays no part. Even if Alice can control the choices of s_a and s_v by manipulating the error rates achieved during the Alice-Bob KGP and the Alice-Charlie KGP, the choice of L takes into account the public parameters s_a and s_v, and the protocol is secure regardless.

6.5 Comparison to QKD

For the finite size, decoy-state BB84 protocol described above, Appendix B of [114] gives the length of the extractable secret key as

$$l \approx s^-_{X,0} + s^-_{X,1}\left[1 - h(\phi^+_{X,1})\right] - \lambda_{EC}, \tag{6.30}$$

where the approximation is due to the omission of some small constants related to the possibility of failure of error correction and privacy amplification. The term λ_{EC} represents the information leaked to Eve during error correction. It depends on the specific implementation of the error correction process, but, according to the CQSW theorem (in Section 3.4.2), must be greater or equal to $nh(e^+_X)$, where n is the size of the bit string being corrected. In practice, error correction will not be perfect and it is common to write $\lambda_{EC} = nf_{EC}h(e^+_X)$ where f_{EC} is a leakage parameter. To perform error correction, the total raw key is split into blocks and the leakage parameter, f_{EC}, depends on this block size, but not the overall length of the key.

Increasing the block size reduces f_{EC} at the cost of decreasing the efficiency of the error correction protocol. Estimates of f_{EC} for practically feasible error correction is an area of active research [119], though it is commonly estimated to be in the range $1.1 - 1.2$, regardless of the length of the total key being distilled. For example, [120] assumes $f_{EC} = 1.2$ based on the performance of error-correcting codes in use at ID Quantique. Rewriting (6.30), we obtain

$$l \approx n \left\{ c^{-}_{X,0} + c^{-}_{X,1} \left[1 - h(\phi^{+}_{X,1}) \right] - f_{EC} h(e^{+}_{X}) \right\}. \tag{6.31}$$

Comparing equations (6.18) and (6.31), we immediately see that the inclusion of f_{EC} means that there are Alice-Bob and Alice-Charlie quantum channels for which quantum USS schemes are possible and yet practical QKD gives a zero key generation rate. As stated above, f_{EC} is independent of n and so cannot be decreased by simply increasing the size of the total key. The important point is that because our quantum USS scheme omits the inefficient process of error correction, in practice there is always some region in which quantum signature generation is possible but secure key distillation is not.

6.6 Experimental implementations

We use this section to outline various experimental implementations of the AWKA scheme. In Section 6.6.1 we simulate the AWKA scheme using realistic system parameters taken from an existing BB84 QKD setup described in Ref. [121]. The simulation allows us to estimate the efficiency of the scheme and to compare it to previous signature experiments.

In Section 6.6.2 we outline the results of two actual implementations of our scheme performed over 90km of installed optical fibre [122, 123]. Both are implementations of the AWKA scheme, but neither use a BB84-type system to perform the KGP. Instead, the KGP is performed using the existing differential phase shift QKD (DPS-QKD) system developed for use in the Tokyo QKD network.

6.6.1 Simulation

In this section we use system parameters taken from Ref. [121] to estimate the number of states that Bob and Charlie need to transmit over a 50 km quantum channel in order to securely sign a 1-bit message over 50 km. We stress that the analyses performed in this example have not been optimised. Instead, it is meant to

illustrate the protocol and to provide approximate signing rates/signature lengths.

Experimental parameters

The experiment uses a 1 GHz source capable of transmitting at three different intensities $(u_1, u_2, u_3) = (0.425, 0.0435, 0.0022)$. The intensities are chosen with probabilities $p_{u_1} = 0.25$, $p_{u_2} = 0.4$ and $p_{u_3} = 0.35$. Independently, the encoding bases are chosen with probabilities $p_X = 0.5$ and $p_Z = 0.5$.[7] The signals are transmitted via optical fibre at 1550 nm achieving a channel attenuation of 0.2 dB/km. The receiver loss at Alice is 2.8 dB and her detectors have efficiency $\eta_{\text{det}} = 20.4\%$. The dark count rate, p_d, is the rate at which the detectors register counts in the absence of any incident light. Here we set $p_d = 2.1 \times 10^{-5}$. Lastly, there is a biased optical bit error rate of $Q_X = 1.38\%$ in the X basis and $Q_Z = 0.76\%$ in the Z basis.

Source and channel estimates

Over 50km, the signal attenuation due to the combined channel and receiver loss is $\eta_{\text{ch}} = 0.0525$. The parameter η represents the overall system transmission, where $\eta = \eta_{\text{det}}\eta_{\text{ch}} = 0.0107$. The detection rates, R_{u_i}, for signals with intensity u_i can be modelled as [124]

$$R_{u_i} = 1 - (1 - 2p_d)e^{-\eta u_i}. \tag{6.32}$$

The X basis bit error rates, e_{X,u_i}, for signals with intensity u_i can be modelled as

$$e_{X,u_i} = \frac{(1 - e^{-\eta u_i})Q_X + e^{-\eta u_i}p_d}{R_{u_i}}, \tag{6.33}$$

and similarly for the Z basis bit error rates.

Protocol parameters and security

A USS scheme is called δ-correct and δ-secure if the probabilities of honest failure, forging, non-transferability and repudiation are all less than δ (see Chapter 4). In what follows we set $\delta = 10^{-4}$. The choice of security level is arbitrary but is chosen to match with the existing quantum USS literature. The security and correctness of the AWKA protocol is described by Eqs. (6.23), (6.25) and (6.29). To evaluate these expressions, we must first set the value of the internal parameters s_a and s_v. From the security proofs above, s_a and s_v must be chosen such that $e_X^+ < s_a < s_v < p_E^*$. If this is not possible, the protocol aborts following the distribution stage.

[7] For longer messages, it would be more efficient to bias these probabilities so that $P_X > 1/2$ and to sign multiple messages using a single Z basis error estimation, as in [111].

The quantity $e^+_{X,I}$, with $I \in \{B, C\}$, is an upper bound on the X basis error rate found from parameter estimation in the KGP performed by Alice and recipient I (see Eq. (6.21)). For this example the Alice-Bob and the Alice-Charlie channels are the same, and as such the recipient subscript is unnecessary. In practice, it is likely that the channels will differ, in which case we set $e^+_X := \max\{e^+_{X,B}, e^+_{X,C}\}$, i.e. we choose the worst case (maximum) of the error rates found in the Alice-Bob and Alice-Charlie KGPs.

Similarly, the quantity $p^*_{E,I}$, with $I \in \{B, C\}$, is a lower bound on the error rate an eavesdropper is able to achieve when dishonestly declaring a signature. The quantity derives from the channel noise estimates and is found using Eq. (6.17). Again, since the Alice-Bob channel could in principle differ from the Alice-Charlie channel, the achievable eavesdropper error rates can also differ. Here, and in all that follows, we set $p^*_E := \min\{p^*_{E,B}, p^*_{E,C}\}$, i.e. we choose the worst case (minimum) of the achievable error rates.

Suppose that each recipient (we focus on Bob, but Charlie will do exactly the same) transmits $T = 6.09 \times 10^8$ states in total[8]. From losses due to the experimental parameters listed, we expect the raw key to contain 2.09×10^5 bit values resulting from successful X basis measurements. Of these, Bob will randomly choose $n = L/2 = 9.94 \times 10^4$ to be B^m_1 and another $L/2$ will be chosen as B^m_2. The remaining $k = 9.94 \times 10^3$ bits make up V_B and will be used to estimate the correlation between Alice's and Bob's X basis measurement outcomes.

For the given intensity probabilities, error rates and detection rates, we expect to observe an X basis bit error rate of 2.87%. Since the overall security level we require is 10^{-4}, we choose $\epsilon_{PE} = 10^{-6}$ meaning $\tilde{\epsilon}_{PE} \leq 1.1 \times 10^{-5}$ (see Appendix A.1). Eq. (6.21) then provides an upper bound on the true error rate as $e^+_X = 5.37\%$.

Finding p^*_E is slightly more involved. The parameters $s_{X,0}, s_{X,1}$ and $\phi_{X,1}$ are estimated using the observed number of errors and counts in different bases/intensities (see Appendix A.1 or Appendix B of [114]). Setting $\epsilon = 10^{-6}$, Eq. (6.2) allows us to estimate the min-entropy as

$$H^{\epsilon}_{\min}(X|E)_{\rho} = 4.12 \times 10^4. \tag{6.34}$$

Together with Eq. (6.17), this gives $p^*_E = 8.36\%$. Note that both e^+_X and p^*_E are found in the distribution stage, and the parameters s_a and s_v are also set in the distribution stage.

[8]This number, though currently obscure, is chosen to provide the required security level of 10^{-4}, as we shall see.

The aim is to choose the internal parameters s_a and s_v so as to maximise the security level for a given signature length. It is important to minimise both the signature length, $2L$, and the number of signals required to generate the signature, T. Reducing L is desirable as it reduces the communication overhead imposed by appending the signature to a message. Reducing T is desirable as it increases the rate at which signatures can be generated. Here we set

$$s_a = e_X^+ + \frac{p_E^* - e_X^+}{4} = 0.0612, \qquad s_v = e_X^+ + \frac{3(p_E^* - e_X^+)}{4} = 0.0761. \tag{6.35}$$

This choice seems reasonable and is in line with previous quantum signature experiments [1]. However, we do not show it is optimal and better choices may exist. Given these parameters, we find

$$\mathbb{P}(\text{Honest failure}) \leq 2.97 \times 10^{-5}, \tag{6.36}$$

$$\mathbb{P}(\text{Forge}) \leq 1.20 \times 10^{-5}, \tag{6.37}$$

$$\mathbb{P}(\text{Repudiation}) \leq 2.98 \times 10^{-5}. \tag{6.38}$$

Results

Overall, the above analysis shows that to sign a 1-bit message to a security level of 10^{-4}, the AWKA protocol requires a signature length of $2L = 1.99 \times 10^5$. This requires the recipients (Bob and Charlie) to transmit 6.09×10^8 states per possible message. For a 1-bit message there are two possibilities, meaning the senders must each transmit 1.22×10^9 states in total. With a 1GHz source, this translates to being able to sign a 1-bit message once every 1.2 seconds[9].

6.6.2 Other experimental implementations

As we have seen, the AWKA scheme uses an underlying QKD-like process (the KGP) to generate correlated keys between participants. Although we have so far chosen a BB84 implementation, we can in fact base the KGP on any valid QKD scheme. Due to its efficiency and ease of use, differential phase shift QKD (DPS-QKD) [125] has become a popular choice of protocol among many experimental groups.

In Refs. [122, 123] the AWKA protocol is performed using a modified DPS-QKD

[9]It should be stressed that this analysis has not been optimised.

link to generate the keys A^m, B^m and C^m. The increased efficiency of DPS systems allows the protocol to be performed over longer distances and requires much shorter keys. The aim of these experiments was to demonstrate the simplicity of performing the AWKA scheme using *any* existing QKD network.

Security

DPS-QKD differs from standard QKD by encoding information into the relative phase of successive pulses. This change allows for simpler experimental implementations, but comes at the cost of reduced security. Until recently there was no proof that DPS-QKD was unconditionally secure, and, in order to make any security statements, it was necessary to place additional restrictions on the adversary's abilities. Unconditional security proofs do now exist [126, 127], but require photon-number-resolving detectors as well as a slightly different setup to the system used in this experiment.

As such, the security analysis performed for this implementation of the AWKA scheme was restricted to adversaries capable of only independent and sequential attacks (i.e. attacks on a limited number of successive pulses). These are the most realistic attacks given current technology, but they do not include all possible attacks. The security against forging attempts relies on results in Ref. [128] to bound the success probability of an eavesdropper attempting to forge a message. The security against repudiation attempts follows similarly to Section 6.4.3, though again, security is only valid for restricted adversaries as above. For full details, see Refs. [122, 123].

Experimental setup

The experimental setup is shown in Figure 6.1 and a full description of the system components is provided in Ref. [123]. For completeness, in this subsection we reproduce the key points.

Only one DPS-QKD system was available, so that states were first sent with the source acting as Bob, and later as Charlie. The transmitting system used a continuous-wave (CW) laser diode with a central wavelength of 1551 nm. The CW output was modulated into a series of pulses using a lithium niobate (LiNbO3) optical intensity modulator driven at clock rate of 1 GHz so that the time between the centre of each pulse was $T = 1$ ns. For each pulse a field programmable gated array (FPGA) selected a phase of 0 or π radians which was subsequently imparted onto the signal using a $LiNbO_3$ phase modulator. The intensity of the optical pulses

was also attenuated to a mean photon number per pulse of 0.2. Additional attenuation was introduced via a variable neutral-density (ND) filter at Bob/Charlie to simulate longer transmission distances. The optical encoder was used for signal synchronisation.

The pulses were transmitted over a 90 km standard telecommunications optical fibre link comprised of a 45 km installed fibre link configured with a loopback at the far end. The total transmission loss was 28.7 ± 0.2 dB, giving a per-unit length loss of 0.32 dB/km. This was used in conversion of the additional attenuation into equivalent length.

Receiver Alice employed a temperature-stabilised silica planar light-wave circuit to introduce a delay of 1 ns so that successive pulses could be interfered. The phase difference between the two successive pulses, either 0 or π, determined which superconducting niobium nitride superconducting nanowire single-photon detector (SNSPD) the pulse was routed towards for detection. This allowed the encoded information to be decoded, with one detector denoted as signifying 0 and the other 1.

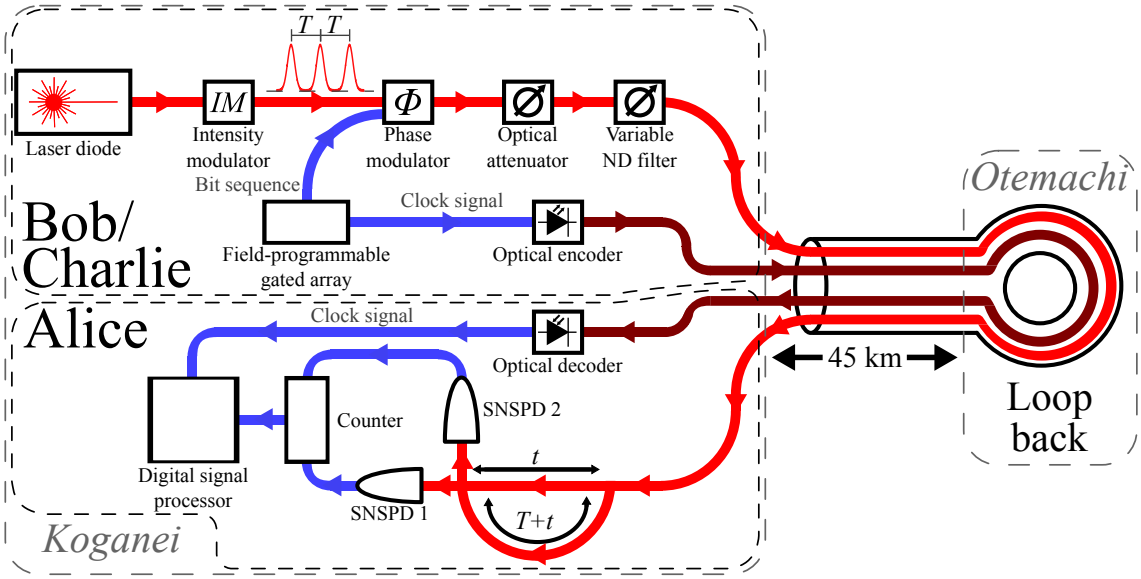


Figure 6.1: The figure, taken from Ref. [123], shows the experimental setup used to perform the AWKA quantum USS scheme.

Results

The system represents a significant advance in the operating length and efficiency of quantum USS systems. Over a distance of 90 km, the system is able to sign a 1-bit message every $t = 0.2$ seconds for a security level of 10^{-4}. The corresponding signature length is $L = 2,502$ bits.

This signature length and generation time improves upon all previous implementations of quantum USS schemes by approximately an order of magnitude. However, as mentioned, these efficiency improvements are partly due to the use of DPS-QKD, and therefore come at the cost of sacrificing the proof of unconditional security.

6.7 Conclusion

In this chapter we have presented a quantum USS protocol and proven its unconditional security against the most general attacks allowed by quantum mechanics. It improves upon all previous quantum USS schemes in a number of ways.

First, it removes all trust assumptions on the quantum channels between participants. Second, despite removing these assumptions, the AWKA protocol also significantly reduces the length of the signature needed to sign a message. The simulation results in Section 6.6.1 suggest that a signature length of $L = 1.99 \times 10^5$ is required to sign a 1-bit message with 10^{-4} security over a distance of 50 km[10]. This would require Bob and Charlie to transmit approximately 6.09×10^8 states (per possible bit to be signed) to Alice during each of their respective KGP's. We compare this to the previously most efficient quantum USS protocols which required both a signature length and the number of states transmitted to be $O(10^{10})$ to achieve 10^{-4} security over just 1 km [1, 59].

The increase in efficiency is largely due to the fact that in the AWKA protocol Bob and Charlie send *different* states to Alice, whereas previous quantum protocols had all been same-state protocols, i.e. Alice sent Bob and Charlie *the same* states. In same-state protocols, even without any eavesdropping, a potential forger has access to a legitimate copy of each of the states Alice sent to the participants. This problem becomes even more serious when generalising to N participants with up to t dishonest parties, since colluding forgers may have t legitimate copies of each state. In our protocol, in which different states are sent by each participant, this problem is evaded. The only source of information for a potential forger is by eavesdropping on the quantum channels – an activity not considered in the theoretical analysis of previous protocols due to the assumption of "tamper-proof" quantum channels.

The third advantage of the AWKA scheme is that it closes the gap between theory and experiment. Previous schemes were proven to be secure within a given theoretical model, but required modifications to make them experimentally viable.

[10]We do not consider the signature length and generation times found in Section 6.6.2, since these are not proven to provide unconditonal security.

These modifications, though small, compromised the security analysis and left the schemes lacking a full security proof against all types of attack. The AWKA scheme allows Bob and Charlie to use a coherent light source, and then decoy-state techniques are used to map the weak coherent states back to the single-photon setting. The result is a protocol that is both provably secure and fully implementable using current technology.

Lastly, we showed that the noise threshold in the quantum channels connecting Alice-Bob and Alice-Charlie is in practice less strict for quantum USS schemes than for distilling a secret key using QKD. For some quantum channels, therefore, USS protocols that use QKD (e.g. P2 of [1]) are not possible, while our direct quantum protocol remains possible. This is a concrete example of a scenario in which direct quantum USS schemes are preferable to classical USS schemes, the latter of which always requires secret shared keys which can only be generated with information-theoretic security using QKD.

Chapter 7

Measurement-device-independent quantum USS schemes

7.1 Introduction

Throughout this thesis we have talked about schemes with information-theoretic security – schemes that are secure as long as the laws of quantum mechanics are true. While the systems considered technically do provide this level of security, it is of a theoretical nature; the security holds within our idealised models of the cryptosystems we are analysing. For instance, a common assumption in both QKD and quantum USS schemes is that participants' labs are completely private from Eve and the outside world. However, in reality it has been shown that there are many reasons why this assumption may not hold true. For example, in any communication protocol, participants' labs must be connected to the channels which link them to other protocol participants. Therefore, their labs are often not completely isolated, and there are open lines through which the eavesdropper can penetrate to gain additional information on the state of the supposedly "private" labs. Loopholes such as these can be very serious, and have been exploited to completely break the security of real-world QKD systems [129–134].

The problem we are describing is one of *side information* – it is possible that the adversary could have additional information, not included in the theoretical model, which allows her to perform powerful strategies which bypass the security proofs. The additional information may arise as a result of a breakdown of explicitly stated assumptions, as suggested above, but more generally can arise in any number of more subtle ways. For example, in the real world, it is highly likely that protocol participants will not create all of the equipment contained within their own lab.

Even if they did, they would usually not be able to guarantee there has been no eavesdropping on the equipment creation process. Much more likely, especially for commercial systems, is that most experimental components – detectors, sources, beamsplitters etc – will be bought from a third party provider who cannot be fully trusted. Any additional information an adversary is able to gain on system components is called side information, and must be characterised and protected against if one wants to claim real-world security. To bridge this new gap between the theory and practice of QKD, there are a number of proposed solutions. As we shall see, these solutions can also be applied to quantum USS schemes.

For QKD, one approach is to attempt to model all possible side channels and prove that the system remains secure against all attacks using the additional side information [93]. This solution seems very difficult, as it requires complex real devices to be modelled and fully characterised. Further, the class of strategies available to the adversary using side information is huge, and characterising them to prove security is a daunting prospect. A perhaps better approach is what's called "device independence". Device independent QKD (DI-QKD) does not require any modelling of any of the devices used in the protocol. Instead, the violation of Bell inequalities is used to infer that the state held by Alice and Bob is close to a maximally entangled state, and from there the monogamy of entanglement can be used to prove security. The major advantage of this approach is that security rests entirely on the correlation statistics observed by Alice and Bob, since it is only these statistics that are used to deduce their shared state. The devices used to generate the statistics need not be trusted as long as the holders are free to choose their measurements and the detection efficiency is sufficiently high. The major disadvantage of this approach is efficiency; because of the detection loophole, high detection efficiency is required. Even then, DI-QKD generates extremely low key rates [135, 136]. Is it possible to use the ideas of device independence but somehow maintain efficiency?

In this chapter we begin in Sections 7.2 and 7.3 by introducing the concept of measurement-device independence; a frontrunner in the potential solutions to the side information problem in the context of QKD. We will see that measurement-device-independence partially bridges the gap between theory and experiment, and provides a higher level of practical security while also maintaining protocol efficiency. In Section 7.4 we follow Ref. [137] in applying these techniques to create the first measurement-device-independent quantum USS protocol. We go on to analyse its security and provide simulation results to demonstrate its efficiency.

7.2 Measurement-device-independent QKD

Historically, detectors have been the most vulnerable part of practical QKD setups. Indeed, the hacking attacks cited above all exploit different detector imperfections. Motivated by this, the goal of measurement-device-independent QKD (MDI-QKD) [138] is to remove all *detector* side-channels. Of course, this does not address other potential side channels such as attacks using source side-channels, and so the aims of MDI-QKD are more limited in scope than fully device-independent QKD. Nevertheless, an efficient MDI-QKD scheme would be an important step forward as it removes almost all known hacking attacks using side information.

The essential idea of MDI-QKD derives from a time-reversed EPR protocol for QKD suggested in 1996 [139]. The time-reversed protocol is very similar to the entanglement based EPR BB84 protocol described in Section 3.3.1, but, unsurprisingly, acts in reverse. In EPR schemes, a maximally entangled state is prepared and later projected onto the BB84 states to generate the key. The key is secure because the eavesdropper does not know what basis Alice and Bob will choose, and only the maximally entangled state can produce perfect correlation in both bases. Monogamy of entanglement then implies security. On the other hand, in reverse EPR schemes the BB84 states are prepared and later projected onto one of the Bell states (which are all maximally entangled) by means of an untrusted party's (Eve's) measurement. The measurement results are announced and used by Alice and Bob to deduce the other's bit, e.g. if Bob sends $|0\rangle$ and Eve announces $|\psi^+\rangle = 1/\sqrt{2}(|01\rangle + |10\rangle)$ as the measurement outcome, Bob can deduce that his bit must be anti-correlated with Alice's (i.e. she sent $|1\rangle$ to Eve) and so flip his bit accordingly (assuming they also post-select on matching basis choices). Security follows because, similarly to before, Eve does not know Alice's and Bob's basis choice, and the only measurement that will produce results which do not lead to errors (between Alice's and Bob's key) is the honest Bell measurement. However, if Eve performs the Bell measurement, she effectively projects Alice and Bob into sharing a maximally entangled state. As before, monogamy of entanglement means that Eve can then have no information on their shared key.

In the reversed EPR scheme, Alice and Bob only need to be able to prepare BB84 states, which are then sent to Eve for measurement. Thus, the measurement device is completely untrusted and it is unnecessary to attempt to characterise it. Despite this, the protocol remains secure! Nevertheless, MDI-QKD does require Alice and Bob to characterise the states they prepare, and this characterisation should take

place in a protected environment outside the influence of the adversary. MDI-QKD combines the idea of time-reversed QKD with decoy-state QKD to produce an efficient, practical and much more secure protocol.

7.3 BB84 MDI-QKD

In this section we describe a decoy-state BB84 MDI-QKD protocol, taken from Ref. [140], which we will later use to construct a measurement-device-independent quantum USS scheme, similar to the AWKA scheme presented in Chapter 6.

1. **State preparation.** Alice chooses a bit value $r \in \{0, 1\}$ uniformly at random and encodes it into a phase-randomised coherent state with three possible intensities – a signal intensity, a_s, and two decoy intensities, a_{d_1} and a_{d_2}. The bit is encoded using either the X or Z basis. The intensity level and encoding basis are chosen randomly by Alice, each with probability $p_{a,\alpha}$, where $a \in \{a_s, a_{d_1}, a_{d_2}\}$ and $\alpha \in \{X, Z\}$. Bob does exactly the same, independently to Alice.

2. **State distribution.** Alice and Bob send their state to Eve using a quantum channel.

3. **Measurement.** If Eve is honest, she makes a Bell state measurement on the received signals. Whether Eve acted honestly or not, she informs Alice and Bob of whether or not her measurement was successful. If successful, she declares the Bell state obtained as the measurement outcome.

4. **Sifting.** If Eve reports a successful result, Alice and Bob communicate their intensity and basis settings using an authenticated classical channel. For each Bell state k, we define two groups of sets: $\mathcal{Z}_k^{a,b}$ and $\mathcal{X}_k^{a,b}$. The sets group the signals according to basis choice (if Alice and Bob choose different bases the signals are discarded), and further by the chosen intensity levels and measurement outcome. The a, b superscript denotes the intensity chosen by Alice and Bob respectively, and k denotes the Bell state measurement outcome declared by Eve. Steps 1–4 are repeated until $|\mathcal{Z}_k^{a,b}| \geq M_k^{a,b}$ and $|\mathcal{X}_k^{a,b}| \geq N_k^{a,b}$ for all a, b and k. The choice of $M_k^{a,b}$ and $N_k^{a,b}$ will depend on the post-processing techniques used and the desired security level. After this, Bob modifies his bits according to the declared measurement outcome to correctly correlate them with those of Alice. The modifications necessary are shown in Table 7.1.

5. **Parameter estimation.** Alice and Bob together choose n_k random bits from $\mathcal{X}_k^{a_s,b_s}$ to form the bit strings X_k held by Alice, and X'_k held by Bob. The remaining R_k bits from $\mathcal{X}_k^{a_s,b_s}$ are used to compute the error rate, $E_k^{a_s,b_s} = \frac{1}{R_k}\sum_l r_l \oplus r'_l$, where r_l and r'_l are Alice's and Bob's bits, respectively. After this the bits in R_k are discarded. If $E_k^{a_s,b_s} > E_{\text{tol}}$ for all k, then Alice and Bob abort the protocol. If $E_k^{a_s,b_s} \leq E_{\text{tol}}$, Alice and Bob use $\mathcal{Z}_k^{a,b}$ and $\mathcal{X}_k^{a,b}$ to estimate $s^-_{k,0}$, $s^-_{k,1}$ and $\phi^+_{k,1}$. The parameter $s^-_{k,0}$ is a lower bound for the number of bits in X_k where Alice sent a vacuum state. Similarly, $s^-_{k,1}$ is a lower bound for the number of bits in X_k arising from when Alice and Bob both sent a single-photon state. $\phi^+_{k,1}$ is an upper bound for the single-photon phase error rate. If $\phi^+_{k,1} > \phi_{\text{tol}}$, the corresponding strings X_k and X'_k are discarded.

6. **Error correction.** For those k that passed the parameter estimation step, Bob obtains an estimate $\tilde{X}_k$ of X_k using an information reconciliation scheme. For this, Alice sends him $\lambda_{\text{EC},k}$ bits of error correction data.

7. **Privacy amplification.** If k passed the error correction step, Alice and Bob apply a random universal hash function to X_k and $\tilde{X}_k$ to extract two shorter strings with higher secrecy. The concatenation of these strings for all non-aborted k values forms the secret key, S.

Alice's & Bob's basis	Bell state reported by Eve			
	$\|\psi^-\rangle$	$\|\psi^+\rangle$	$\|\phi^-\rangle$	$\|\phi^+\rangle$
Z	Bit flip	Bit flip	–	–
X	Bit flip	–	Bit flip	–

Table 7.1: Processing of data in the sifting stage. The Bell states are defined as $|\psi^-\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle), |\psi^+\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle), |\phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ and $|\phi^-\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle)$.

MDI-QKD security

A central aim of any QKD protocol analysis is to find the maximum length of the generated key, S, held by Alice and Bob, such that S can be proven to be almost perfectly secret (as per Definition 3.3). A crucial element in finding the length of the generated key is expressing Eve's uncertainty on the sifted key X_k (before error correction and privacy amplification) in terms of her min-entropy. For the protocol above

$$\begin{aligned} H_{\min}^{\epsilon_k}(X_k|E) &\geq s^-_{k,0} + s^-_{k,1}[1 - h(\phi^+_{k,1})] - 2\log_2 \frac{2}{\epsilon'_k \tilde{\epsilon}_k} \\ &\gtrapprox s^-_{k,0} + s^-_{k,1}[1 - h(\phi^+_{k,1})], \end{aligned} \tag{7.1}$$

where $\epsilon_k \geq \epsilon'_k + \tilde{\epsilon}_k$. The approximation on the second line is valid because the logarithmic term is small compared to the preceding two terms. For clarity, and since it does not impact our later results, we omit explicit references to the logarithmic term.

Advantages of MDI-QKD

As discussed above, the major advantage of MDI-QKD is that it removes all possible detector side-channel attacks, thus bringing theory further in line with practical implementations. On top of this, the scheme also enjoys a number of secondary advantages discussed below.

First, a severe practical limitation of all QKD schemes is that they are fundamentally distance limited – current fibre-based QKD systems are restricted to distributing key over distances up to approximately 250km, but for efficiency typically operate at distances of less than 100km. Theoretical results show that this limitation is inherent to any optical QKD scheme [141] and cannot be overcome without quantum memory. By placing the detectors halfway between Alice and Bob, MDI-QKD effectively doubles the achievable transmission distance.

Second, MDI-QKD is very efficient compared to other attempts to remove side-channel attacks. Fully DI-QKD suffers hugely from problems associated with the detection efficiency loophole, which requires an overall detection efficiency of around 80%. For practical QKD setups in which there is high channel loss and only imperfect detectors available, DI-QKD becomes essentially impossible, with expected secret key rates falling below 1 bit per second (bps) if possible at all. On the other hand, recent advances in experimental techniques have allowed MDI-QKD systems to achieve secret key rates of 9.7×10^4 bps over a distance of 52km, and Mbps secret key rates over shorter distances [142]. These rates are even comparable to the state-of-the-art measurement-device-*dependent* QKD systems.

Third, MDI-QKD removes the need for either Alice or Bob to have detectors. Detectors are often the most expensive and complex element of a QKD system, and could significantly increase the cost of purchasing/maintaining a QKD link between two parties. MDI-QKD allows for the possibility of an untrusted central node holding all measurement equipment and connecting many parties. From a commercial perspective, this could be very beneficial in larger networks since it reduces the cost for each individual Alice and Bob. Instead, they could use third party measurement providers, such as Eve, whom they do not even need to trust.

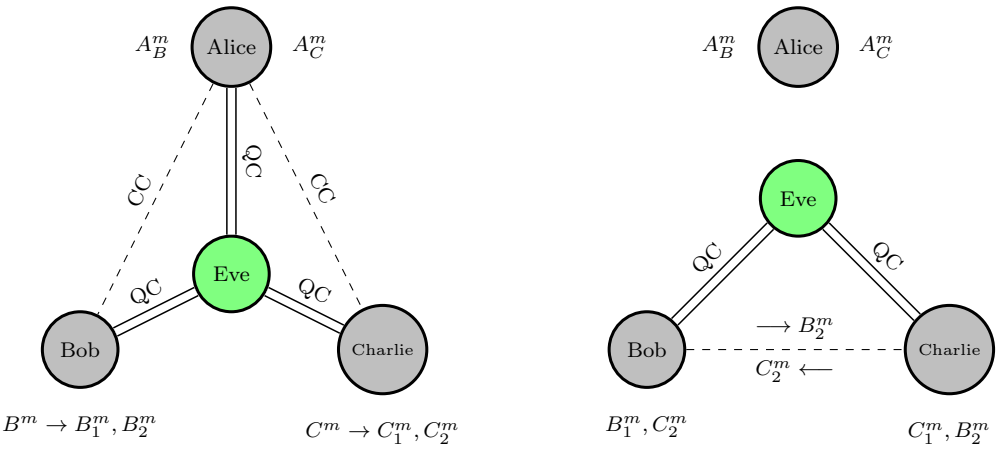
7.4 MDI quantum USS schemes

Just as QKD can suffer from detector side-channel attacks, so too can quantum USS schemes. In the context of the AWKA scheme presented in Chapter 6, Eve could, for example, employ a detector hacking strategy to produce a string E_{guess} such that $d(A_B^m, B_m) \geq d(E_{\text{guess}}, B_m)$ with probability greater than ϵ (c.f Eq. (6.1)) even when the channel noise is low. This is not due to a flaw in the security proof, but rather due to the limitations of the model assumptions – namely, our model assumes Alice's and Bob's labs are completely secure, and does not consider potential side-channel attacks. Detector hacking strategies therefore fall outside of the scope of the strategies included in the supremum in Eq. (6.1).

More generally, any quantum USS scheme will be vulnerable to detector side-channel attacks since all involve the transmission and measurement of quantum states. Perhaps due to the relative immaturity of quantum USS schemes compared to QKD, together with the lack of any "standard" USS scheme, side-channel attacks have not been considered in the USS literature. In general, removing potential side channels from quantum USS schemes is a tough open problem, and for many schemes it is not clear how to achieve measurement-device independence, or even whether it is possible. However, a major benefit of the AWKA scheme is its similarity to QKD, which means that the concepts from MDI-QKD can be directly applied to create the first MDI quantum USS scheme. The results presented in this section have been published in Ref. [137].

7.4.1 The MDI-AWKA protocol

In this section we modify the AWKA protocol described in Section 6.2 to make it fully measurement-device-independent. The idea is simple: the protocol proceeds in exactly the same way as before, except that whenever Alice, Bob or Charlie need to perform either the KGP or QKD, they instead perform an MDI version using an untrusted party, Eve, to perform all measurements. The modified distribution stage is shown in Figure 7.1, while the messaging stage remains unchanged. For a detailed description of each stage, refer to Section 6.2 and note that whenever the KGP or QKD is mentioned, here it is replaced by a measurement-device-independent version (the MDI-KGP is described below). Finally, recall that for signature protocols, in which any party could be dishonest, Eve could actually be Alice, Bob or Charlie.



(a) MDI-KGP. (b) MDI-QKD.

Figure 7.1: The distribution stage of the MDI-AWKA protocol. In (a), Alice, Bob and Charlie all have quantum channels to Eve. Alice-Bob and Alice-Charlie are also connected via authenticated classical channels. Similarly to the AWKA protocol, they use these channels to perform the MDI-KGP to generate the sets A_B^m, B^m, A_C^m and C^m, independently for each of $m = 0, 1$. Bob and Charlie then randomly (and secretly) split their sets, B^m and C^m, in half. In (b), Bob and Charlie use the quantum channels to Eve, together with the Bob-Charlie authenticated classical channel, to perform MDI-QKD to create a secret classical channel. They use this to transmit B_2^m and C_2^m to the each other in secret.

The MDI-KGP

The KGP is simply the quantum part of QKD without the classical post-processing steps of error correction and privacy amplification. Similarly, the MDI-KGP is simply steps 1–5 of the MDI-QKD protocol described above, with steps 6 and 7 omitted. For example, to perform the MDI-KGP to generate sets A_B^m and B^m, Alice and Bob each send a sequence of phase-randomised weak coherent pulses to Eve, who announces a Bell state as the measurement outcome. Alice and Bob sift their results to filter out any positions where they chose different bases, and they perform the correction operations specified by Eve's measurement outcome. The resulting string held by Alice is A_B^m and the string held by Bob is B^m. They do not perform error correction or privacy amplification, so the strings will be neither identical nor perfectly secret. However, as long as the error rate found in parameter estimation is sufficiently low, we will show that the MDI-KGP is still ϵ-secure, i.e.

$$\sup\left\{\mathbb{P}\Big(d(A_B^m, B^m) \geq d(E_{\text{guess}}, B^m)\Big)\right\} \leq \epsilon, \tag{7.2}$$

where the supremum is taken over all strategies for Eve allowed by quantum mechanics. As before, the probability is taken over E_{guess}, Eve's attempt at guessing B^m, and $d(.,.)$ is the Hamming distance. Note that now, since neither Alice nor Bob have detectors, there can be no detector side-channel attacks available to Eve.

MDI-KGP security

Suppose that Alice and Bob perform the MDI-KGP so that Bob generates the strings $(V_B, Z_B, X_{B,\text{keep}}, X_{B,\text{forward}})$ for use in the signature protocol, exactly as described in Section 6.3. As before, the string $X_{B,\text{keep}}$ has length n and denotes the outcome set B_1^m. Similarly, the string $X_{B,\text{forward}}$ also has length n and denotes the outcome set B_2^m [1]. The strings V_B and Z_B are used to estimate channel error rates and then they are discarded. The MDI-QKD results stated above in Eq. (7.1), together with very similar arguments to those in Section 6.3.2, lead to

$$H_{\min}^{\epsilon}(X_{B,\text{keep}}|E) \gtrapprox s_0^- + s_1^-[1 - h(\phi_1^+)], \tag{7.3}$$

where s_0^- is a lower bound for the number of bits in $X_{B,\text{keep}}$ where Alice sent a vacuum state, s_1^- is a lower bound for the number of bits in $X_{B,\text{keep}}$ where Alice and Bob sent a single-photon state, and ϕ_1^+ is an upper bound for the single-photon phase error rate.

Once the conditional min-entropy is known, Theorem 6.1 can be used to bound p_r, Eve's probability of making at most r mistakes when guessing $X_{B,\text{keep}}$. As before,

$$p_r \leq \sum_{j=0}^{r} \binom{n}{j} 2^{-H_{\min}^{\epsilon}(X_{B,\text{keep}}|E)} + \epsilon. \tag{7.4}$$

This bound is used to prove security against forging in the full signature protocol.

7.4.2 The MDI-AWKA protocol security

With the security of the MDI-KGP given above, security of the full MDI-AWKA protocol proceeds similarly to the analysis performed in Section 6.4. Below we summarise the arguments.

Robustness

Parameter estimation on the V strings generated during the MDI-KGP leads to an observed error rate (with Alice's signature (A_B^m, A_C^m)) of $\tilde{e}_{X,B}$ for Bob and $\tilde{e}_{X,C}$ for Charlie. Serfling's inequality allows us to upper bound the actual error rate by $e_{X,B}^+$ and $e_{X,C}^+$, as per Eq. (6.21). These bounds hold except with probability ϵ_{PE}. Setting $e_X^+ := \max\{e_{X,B}^+, e_{X,C}^+\}$ and choosing s_a such that $s_a > e_X^+$, we find that in

[1] As in Chapter 6, for the sake of clarity we duplicate notation, i.e. set $X_{B,\text{keep}} = B_1^m$ and $X_{B,\text{forward}} = B_2^m$.

the honest case Bob will accept Alice's signature except with probability

$$\mathbb{P}(\text{Honest Failure}) \leq 2\epsilon_{PE}. \tag{7.5}$$

Forging

Suppose Bob is trying to forge a message to Charlie. Parameter estimation provides a bound on the maximum error rate between A_C^m and C_1^m, call it e_X^+. As before, Theorem 6.1 and Eq. (6.17) can be used to bound, p_E^*, the minimum rate at which Bob/Eve can make errors, as

$$\frac{1}{n}H_{\min}^{\epsilon}(X_{B,\text{keep}}|E) \leq h(p_E^*). \tag{7.6}$$

Assuming $e_X^+ < p_E^*$ (if not, the protocol is aborted), we choose s_v such that $e_X^+ < s_v < p_E^*$ and find

$$\mathbb{P}(\text{Forge}) \leq 2^{-\left\{H_{\min}^{\epsilon}(X_{B,\text{keep}}|E) - nh(s_v)\right\}} + \epsilon + \tilde{\epsilon_{PE}}, \tag{7.7}$$

where $\tilde{\epsilon_{PE}} > 0$ is the probability of failure of any of the upper/lower bounds on the estimated quantities e_X, s_0, s_1 and ϕ_1.

Repudiation

Security against repudiation derives from the key exchange performed by Bob and Charlie over a secret classical channel. This part of the protocol is unaffected by the switch to measurement-device independence since Bob and Charlie still perform full QKD. Therefore, as before, we choose $s_v > s_a$ and find

$$\mathbb{P}(\text{Repudiation}) \leq 2\exp\left[-\frac{1}{2}(s_v - s_a)^2 n\right]. \tag{7.8}$$

Overall, we see that the protocol is correct and that the probability of forging, repudiating, or non-transferability decays exponentially with the length of the signature.

7.4.3 Advantages of MDI-USS schemes

The advantages of MDI-USS schemes are the same as the advantages enjoyed by MDI-QKD. First and foremost, it brings the theory further in line with practice by removing detector side-channel attacks. To varying degrees MDI-USS schemes also

enjoy the same secondary benefits such as increased transmission distances, an only moderate efficiency loss, and a decrease in the required number of physical quantum channels.

Unlike QKD, USS schemes necessarily involve $N > 2$ parties meaning MDI schemes allowing all participants to simply be connected via quantum channels to a central untrusted node is a potentially major advantage, more so than for QKD. On the other hand, the loss in efficiency arising from measurement-device independence, though moderate, is particularly damaging to USS schemes since inefficiency is already their major drawback.

7.4.4 Simulation results

In this section we present the results of the simulation performed in Ref. [137] to estimate the number of quantum transmissions necessary to sign a 1-bit message to a security level of 10^{-4} and 10^{-10} over 50 km. The analysis closely resembles the one presented in Chapter 6 of this thesis, and is not repeated here. For further details, see the Appendices A and D from Ref. [137].

Using realistic experimental quantities, the simulation finds that a signature length of $2L = 3.56 \times 10^7$ will suffice for a security level of 10^{-4}. This requires Bob and Charlie to transmit a total of approximately $N_{\text{sig}} = 2.23 \times 10^{13}$ quantum states to Eve to perform the two (each) required MDI-KGPs (one each for each possible future message $m = 0$ or 1). Using a 1 GHz source we calculate that it would take approximately 372 minutes to perform the distribution stage when the experiment uses single-photon detectors with a detection efficiency (η_D) of 14.5%. Of course, detectors with higher efficiency will reduce the signature generation time.

Table 7.2 shows the signature generation times for various existing detectors which could be used in the protocol. The most advanced superconducting nanowire single-photon detectors (SNSPDs), which have a 93% efficiency [143], require Bob or Charlie to send 2.56×10^{11} signals to generate the signature, which could be done in 6.4 minutes. The table also shows the signature generation times if a security level of 10^{-10} is used instead.

Clearly, the signature generation times are currently too long for the scheme to be considered practical. Nevertheless, the scheme presented in this chapter is the first MDI quantum USS scheme and should be considered as a proof of concept, first iteration scheme. Since it can be performed using the same equipment as required by QKD (with only minor modifications), we believe that many experimental and theoretical improvements exist allowing the scheme to become much more efficient

Detectors	$\eta_D(\%)$	$Y_0(\times 10^{-6})$	$N_{\text{sig}}(\times 10^{12})$		t_r(min)	
			10^{-4}	10^{-10}	10^{-4}	10^{-10}
Standard [144]	14.5	6.02	22.3	42.1	372	700
InGaAs APD [145]	30	130	7.20	13.4	120	223
InGaAs/InP APD [146]	55	500	3.48	6.52	58	108
SNSPDs [143]	93	1	0.392	0.72	6.4	12

Table 7.2: Raw key generation times for various detectors that could be used in a MDI-USS protocol for a distance of 50 km and security thresholds of 10^{-4} and 10^{-10}. The parameters $\eta_D(\%), Y_0$ and N_{sig} denote respectively the detection efficiency, dark count rate of Eve's detectors, and the number of signals that Bob/Charlie sends to Alice during their KGPs. t_r is the time taken to generate the raw key assuming a source with a pulse rate of 1 GHz.

as well as remaining implementable with current technology.

Indeed, a recent paper [142] employs an existing MDI-QKD setup to perform the MDI quantum USS protocol described here, and is able to generate a 1-bit message signature every 45 seconds to a security level of 10^{-10}. The speedup is gained despite the fact that the detectors used are InGaAs APDs with an average efficiency of just 20.9%. The significant improvement in the signature generation time can be explained partly by a better optimisation of system parameters, but is mainly due to certain "economies of scale" that appear when using the protocol to generate more than one signature. Namely, the system was run for a prolonged length of time – sufficient to collect enough key to generate $2,506$ independent 1-bit signatures. When many signatures are generated from a large block of collected data, the authors show that a single estimation procedure is sufficient to characterise the channel/eavesdropping information for all signatures created from that block. This leads to an order of magnitude decrease in the average number of required signals per 1-bit message signature.

7.5 Conclusion

In this chapter we have introduced the first MDI quantum USS scheme and proven it unconditionally secure. The scheme helps to further bridge the gap between theory and real-world implementations by removing all detector side-channels, thus ruling out a wide class of potential hacking attacks. The protocol implementation only requires participants to send coherent states to a central untrusted node who performs a Bell state measurement. This similarity to QKD means that MDI quantum USS schemes could easily be deployed in existing QKD networks with only small overheads, as demonstrated in Ref. [142]. The MDI structure could also reduce the cost of USS schemes over larger networks since it both removes the need for participants

to hold detectors, and reduces the number of physical quantum channels required. On the other hand, MDI quantum USS schemes suffer from a moderate reduction in protocol efficiency which, for signatures, further reduces already impractical signing rates. Nevertheless, since research into MDI quantum USS schemes is still in its infancy, we expect there could be many theoretical and experimental advances which could significantly improve the quoted signing rates.

Chapter 8

The hash scheme

8.1 Introduction

The previous two chapters have focused on the development of practical *quantum* USS schemes. In this chapter we propose a new *classical* USS scheme, referred to as the "hash scheme", which naturally extends the unconditionally secure MACs introduced in Section 3.6. The main difference between an unconditionally secure MAC and an USS scheme is that signature schemes ensure the transferability of signed content, while authentication codes do not. We propose a method, similar to secret sharing [147], allowing unconditionally secure MACs to be transformed into classical USS schemes[1]. In the hash scheme, a sender shares with each of the remaining protocol participants (or recipients) a set of keys (hash functions) from a family of universal hash functions. The recipients then share with each other a random portion of the keys that they received from the sender. A signature for a message is a vector of tags generated by applying the hash functions to the message. As for MACs, the practical implementation of the hash scheme is straightforward and efficient.

There were two main motivations for this chapter. First, all realisable quantum USS schemes and many classical USS signature protocols are Lamport-type schemes, in which participants must perform the distribution stage many times to sign a single future message. Effectively, the distribution stage is performed once for each possible future message. In order to sign longer messages, this strategy is hugely inefficient – to sign an arbitrary n-bit message as a whole, the distribution stage needs to be performed 2^n times; alternatively, if the message is signed bit-by-bit, the distribution stage would need to be performed $2n$ times (twice for each bit in the message), and

[1]The hash scheme can be thought of as a transferable MAC.

one would need to be careful as to how overall security was defined. The scheme proposed in this chapter aims to resolve and remove this inefficiency. Second, we have seen that quantum schemes are partly motivated by their ability to seemingly provide USS schemes requiring fewer resources than their classical counterparts. This motivation does not always hold, and for example classical USS schemes such as P2 [1] exist using only the resources contained in the standard resource model. Nevertheless, we aimed to find a classical USS scheme that further reduced the resources required to sign a message in order to further explore whether requiring fewer resources was a true advantage of quantum USS schemes.

As we shall see, the scheme we present addresses both points above, as well as others. Using the distribution stage to send hash functions (rather than bit values as in previous schemes) allows participants to sign *any* message (up to a given maximum size) using just the single distribution stage. Further, the scheme achieves this significant efficiency boost while using *fewer* resources than quantum USS schemes; namely, the hash scheme only uses resources scaling similarly to authenticated classical channels (see Section 8.4).

Compared to the most efficient realisable quantum USS scheme, the hash scheme is a huge improvement when considering larger messages. For 51 participants signing a 1 Mb message, both the secret key required by each participant and the signature length is reduced by a factor of at least 10^6 (see Section 8.6.2). The disparity in size becomes larger as the message size increases. A further advantage of classical schemes over quantum schemes is simplicity – implementation of classical schemes is easier as it does not necessarily require quantum state preparation/detection[2]. As such, if quantum USS schemes are to compete, they must provide additional motivation for their use.

Direct comparisons of our new protocol to existing classical USS schemes are more difficult due to the variety of different resources assumed in each. Nevertheless, as we shall see in Section 8.6, even compared to the most efficient and practical classical USS schemes, the hash scheme enjoys a number of favourable properties such as short secret key requirements, short signature lengths, and high computational efficiency. Our contributions and the chapter outline can be summarised as follows.

- We construct a classical USS scheme that, unlike most prior schemes, does

[2]The removal of quantum state transmission also means that classical schemes are not necessarily distance limited. However, this latter point is not entirely fair, since all classical schemes require a secret shared key, and these can only be generated with information-theoretic security using QKD.

not rely on a trusted authority, detectable broadcast or anonymous channels (Section 8.2).

- We prove the information-theoretic security of our scheme against forging, repudiation, and non-transferability (Section 8.3).
- We show that the resources required by our scheme are minimal and have the same scaling as message authentication (Section 8.4).
- Although our scheme does not rely on trusted third parties, we show that having a trusted authority makes our scheme even more attractive (Section 8.5)). In addition, we discuss other possible extensions to our scheme.
- We compare our schemes with existing classical and quantum USS schemes, as well as some common quantum-safe signature schemes (Section 8.6). The comparisons show that the hash scheme has a number of unparalleled advantages over the previous USS schemes.

The work presented in this chapter is taken from Ref. [148] with minor modifications.

8.2 The protocol

The hash scheme is inspired by the protocol named Generalised P2 (GP2), first introduced in [1], and subsequently extended and formalised in [34]. However, contrary to GP2, in which participants independently distribute bit values for each possible future message, our new scheme requires participants to distribute universal hash functions (chosen from an ϵ-ASU_2 set) which are later used to sign *any* possible future message (up to a given maximum size).

Almost strongly universal hash functions are used extensively throughout this chapter, and are discussed in Section 3.6.3. The effectiveness of the protocol relies on our ability to find an ϵ-ASU_2 set which is "small" so that participants can exchange the hash functions efficiently. Fortunately, finding small ϵ-ASU_2 sets is an active area of research, and many already exist. In this chapter we will use the following theorem.

Theorem 8.1 ([98]). *There exists an ϵ-ASU_2 set $\mathcal{F} = \{f \mid f : \mathcal{M} \to \mathcal{T}\}$ with $\epsilon = 2/|\mathcal{T}|$, such that if $a := \log|\mathcal{M}|$ and $b := \log|\mathcal{T}|$, then*

$$|\mathcal{F}| = 2^y, \tag{8.1}$$

where $y := 2^{3b+2s}$ *and* s *is defined by the equation* $a = (b+s)(1+2^s)$.

The functions in $\mathcal{F}$ are chosen to map messages in the set $\mathcal{M}$ to tag values in the set $\mathcal{T}$. Accordingly, we refer to $\mathcal{M}$ as the *message set* and $\mathcal{T}$ as the *tag set*. The theorem means that fully specifying an element $f \in \mathcal{F}$ requires y bits, where y depends on both the maximum allowed message length and the tag length.

Protocol overview

The protocol contains $N+1$ participants: a sender P_0 and N receivers, $P_1, ..., P_N$. Before the protocol, all participants agree on an ϵ-ASU$_2$ family of functions, $\mathcal{F}$, where $\epsilon = 2/|\mathcal{T}|$. The basic idea is for the sender to give each recipient a number of keys (hash functions) which will be used in future to authenticate a message by appending tags (hash values) to the message being sent. To check the signature, participants will apply their hash functions to the message, and check that the outcome matches the tags appended to the message by the sender. They will count the number of mismatches between their hash values and the appended tags, and only accept the message if they find less than a threshold amount of mismatches. However, if the sender were to know which hash functions are held by which participant, she could choose to append appropriate tags such that one recipient accepts the message while another does not, thereby breaking transferability of the scheme. To ensure transferability then, each recipient will group the hash functions received from the sender into N equally sized sets (of size k), and send one set (using secret channels) to each of the other $N-1$ recipients, keeping one set for himself. The recipients test each of the N sets independently.

Transferability levels

The situation is further complicated if the sender is in collusion with some of the recipients. In that case, the sender can have partial knowledge on who holds which keys. This forces us to define levels of transferability. Levels of transferability are perhaps confusing, so here we will try to highlight the need for such levels.

Imagine that a sender is in collusion with a single recipient. In this case, the sender knows k of the keys held by honest recipient H_1, and k of the keys held by honest recipient H_2 - namely, he knows the keys that were forwarded to the honest recipients by his dishonest partner. For these known keys, the sender can attach tags that are correct for H_1, and are incorrect for H_2. Therefore, based on the number of colluding adversaries, the sender is able to bias the number of mismatches and the

number of incorrect sets found between each honest party. To ensure transferability then, we require that the second verifier accepts a message as authentic even if each set contains a higher number of mismatches, and there are more invalid sets than found by the first verifier. Of course, to ensure security against forging, we cannot allow message-signature pairs containing too many errors to be accepted, and so there must be a cap on the highest level of mismatches acceptable by anyone. This leads to levels of verification, and a limit on the number of times a message is guaranteed to be transferable in sequence.

For clarity, suppose then there are three levels of verification, l_0, l_1 and l_2. Accepting a message at any of these levels means the message is guaranteed to have originated with the claimed sender. If H_1 accepts a message at level l_2 (the highest verification level, i.e. the level with the fewest errors in the signature), then he can forward it to H_2, who will first try to accept the message at level l_2. If he finds too many mismatches for the message to be accepted at level l_2, he will instead try to verify at level l_1. The protocol ensures that if H_1 found the message to be valid at level l_2, then H_2 will find the message to be valid at level l_1. Therefore, with three verification levels, accepting the message at level l_2 guarantees that the message can be transferred at least twice more. In practice, the message may be transferred many more times, since with honest participants it is highly likely that H_2 will also find the message valid at level l_2 and they will not need to move to the next verification level.

With this in mind, to begin the protocol we must first decide the maximum number of dishonest participants we want our protocol to be able to tolerate (which, as per the preceding paragraph, will impact our verification levels). We set this to be ω such that $\omega < (N+1)/2$, since the protocol cannot be made secure (using the majority vote dispute resolution process) if more than half of the participants are dishonest (see Section 4.3). We also define the notation $d_R := (\omega - 1)/N$, i.e. d_R is the maximum fraction of dishonest *recipients* possible when the sender is part of the coalition.

As in previous protocols, there are two stages - the distribution stage and the messaging stage.

Distribution Stage

1. The sender independently and uniformly at random selects (with replacement) N^2k functions from the set $\mathcal{F}$, where k is a security parameter. We denote these functions by $(f_1, ..., f_{N^2k})$ and will refer to them as the *signature functions*.

2. To each recipient, P_i, the sender uses secret classical channels to transmit the Nk functions $(f_{(i-1)Nk+1}, ..., f_{iNk})$. As per Theorem 8.1, this requires the sender to share Nky secret bits with each recipient.

3. Each recipient P_i randomly splits the set $\{(i-1)Nk+1, ..., iNk\}$ into N disjoint subsets of size k, which we denote $R_{i\to 1}, ..., R_{i\to N}$. These sets form an index of the functions that P_i will forward to the other recipients. Specifically, P_i uses the secret classical channels to send $R_{i\to j}$ and $F_{i\to j} := \{f_r : r \in R_{i\to j}\}$ to recipient P_j. To securely transmit the signature functions and their positions requires each pair of participants to share $ky + k\log(Nk)$ secret bits. Following this symmetrisation, participant P_i holds the Nk functions given by $F_i := \bigcup_{j=1}^{N} F_{j\to i}$ and their positions given by $R_i := \bigcup_{j=1}^{N} R_{j\to i}$. We refer to these as the *key functions* and *function positions* of participant P_i. The participants will use these to check a future signature declaration.

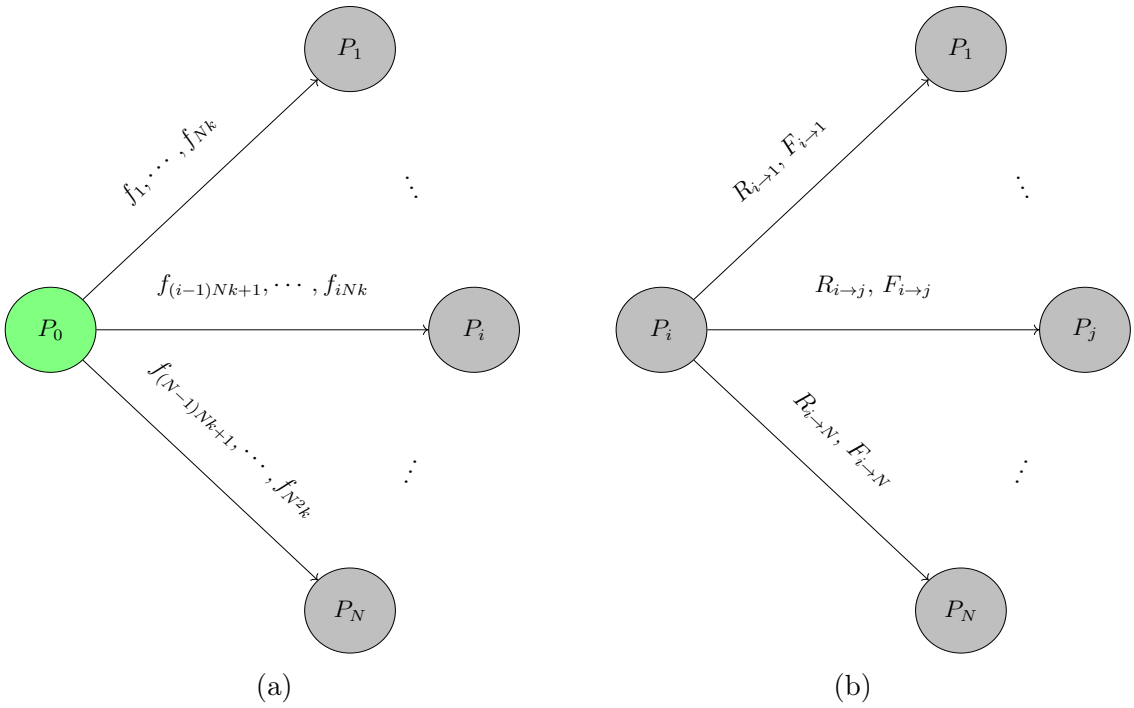


Figure 8.1: The distribution stage of the hash scheme. Figure (a) shows Steps 1 and 2 of the distribution stage, in which the sender P_0 shares distinct sets of keys with all of the receivers $P_1, \cdots, P_N$. Figure (b) shows Step 3 of the distribution stage, in which the recipients exchange a randomly selected portion of their keys with each other.

Messaging Stage

1. To send message $m \in \mathcal{M}$ to P_i, the sender sends (m, Sig_m), where

$$\mathrm{Sig}_m := (f_1(m), f_2(m), \ldots, f_{N^2k}(m)) = (t_1, \ldots, t_{N^2k}).$$

Since the tags have size b, the signature is N^2kb bits in size.

2. For message m and the signature elements t_r such that $r \in R_{j\to i}$, participant P_i defines the following test

$$T^m_{i,j,l} = \begin{cases} 1 & \text{if } \sum_{r\in R_{j\to i}} g(t_r, f_r(m)) < s_l k \\ 0 & \text{otherwise} \end{cases} \tag{8.2}$$

where s_l is a fraction defined by the protocol implementation, such that $1/2 > s_{-1} > s_0 > ... > s_{l_{max}}$, and $g(.,.)$ is a function of two inputs which returns 0 if the inputs are equal, and 1 if the inputs are different. Essentially, this is a test on the set of functions $F_{j\to i}$ to check whether a sufficient number of the tags in the signature match the output of the functions when applied to the message. For each fixed l, if the outcome of the test is 1, we say that that test is passed at level l. For any verification level, the recipient will perform N such tests, one for each $j = 1, ..., N$. Note that participant P_i can perform all of these tests *without* interaction with any other participant.

3. Participant P_i will accept (m, Sig_m) as valid at level l if

$$\sum_{j=1}^{N} T^m_{i,j,l} > N\delta_l \tag{8.3}$$

That is, participant P_i accepts the signature at level l if more than a fraction of δ_l of the tested sets are passed, where δ_l is a threshold given by $\delta_l = 1/2 + (l+1)d_R$.

4. To forward a message, participant P_i simply forwards (m, Sig_m) to the desired recipient.

8.3 Security analysis

8.3.1 Forging

Recall the definition of forging, provided in Definition 4.9. In order to forge, a coalition C (which does not include the signer) with access to a single message-signature pair (m, Sig_m) must output a distinct message-signature pair $(m', \text{Sig}_{m'})$ that will be accepted (at any level $l \geq 0$) by a participant $P_i \notin C$. We consider

forging to be successful if the coalition can deceive any (i.e. at least one) honest participant.

Theorem 8.2. *The protocol defined in Section 8.2 is secure against forging attempts.*

Proof. It is easiest for the coalition to forge a message at the lowest verification level $l = 0$, so we consider this case in what follows. We further assume that the coalition hold a valid message-signature pair (m, Sig_m). We first restrict our attention to the coalition trying to deceive a fixed participant, and we will prove that this probability decays exponentially fast with the parameter k. We then use this to bound the general case where the target is not a fixed participant. Therefore, for now, we fix the recipient that the coalition wants to deceive to be $P_i \notin C$.

To successfully forge, as per Eq. (8.3), the coalition should output a message-signature pair, $(m', \mathrm{Sig}_{m'})$, that passes at least $N\delta_0 + 1$ of the N tests performed by P_i in step 2 of the messaging stage, where $m' \neq m$. Since $\delta_0 = 1/2 + d_R$ and $d_R := (\omega - 1)/N$, this means $N\delta_0 + 1 = N/2 + \omega$. By the definition of the protocol, the number of members in a coalition is at most ω. The coalition knows $F_{d\to i}$ and $R_{d\to i}$ for all $P_d \in C$, so they can use this knowledge to trivially ensure that P_i passes ω of the N tests performed at level $l = 0$. To pass the required $N/2 + \omega$ tests, the coalition must pass a further $N/2$ tests out of the $N - \omega$ remaining tests. The first step in computing the probability that they are able to do this is to calculate the probability of the coalition being able to create a signature such that, for a single $P_j \notin C$, $T^{m'}_{i,j,0} = 1$, i.e. the probability that the coalition can guess the tags forwarded from a single honest recipient P_j to P_i.

Let p_t denote the probability that the coalition can force $T^{m'}_{i,j,0} = 1$, when they have no access to $(F_{j\to i}, R_{j\to i})$, i.e. p_t is the probability that the coalition can create a message-signature pair that will pass the test performed by P_i for the functions received from $P_j \notin C$. As per the protocol, P_j sent $(F_{j\to i}, R_{j\to i})$ to P_i using secure channels, and therefore $F_{j\to i}$ and $R_{j\to i}$ are unknown to the coalition. However, we assume that the coalition possess a valid message-signature pair (m, Sig_m), from which they can gain partial information on $(F_{j\to i}, R_{j\to i})$. Let us denote the k unknown functions in $F_{j\to i}$ by $u_1, ..., u_k$, and consider how the coalition might try to guess the value of $t'_1 := u_1(m')$, given $t_1 := u_1(m)$, where $m' \neq m$.

Since $\mathcal{F}$ is ϵ-ASU_2, using Definition 3.16 the coalition immediately knows u_1 is in a set $\mathcal{F}_1 \subset \mathcal{F}$ which has size $|\mathcal{F}|/|\mathcal{T}|$. Upon receiving message m', P_i will be expecting to find tag t'_1 in the signature. The coalition does not know t'_1 though, so the best they can do is to pick a random function in $\mathcal{F}_1$, and hope that this

function also maps m' to the unknown t'_1. Again by Definition 3.16, the fraction of functions in $\mathcal{F}_1$ that map m' to t'_1 is at most $2/|\mathcal{T}|$. Therefore, the probability that the coalition chooses a function that gives the correct tag for message m' is $2/|\mathcal{T}|$. This is independently true for each of the k unknown functions.

Let X be the random variable that counts how many incorrect tags the coalition declares. Then X follows a binomial distribution and we have

$$p_t = \mathbb{P}(X < ks_0) = \sum_{v=0}^{ks_0-1} \binom{k}{v} \left(\frac{2}{|\mathcal{T}|}\right)^{k-v} \left(1 - \frac{2}{|\mathcal{T}|}\right)^{v}. \tag{8.4}$$

This decays exponentially fast with the parameter k. For example, it may be desirable to choose a small tag length in order to minimise the length of the signature. For $|\mathcal{T}| = 4$ the signature is $2N^2k$ bits in size and we have

$$p_t = \sum_{v=0}^{ks_0-1} \binom{k}{v} \left(\frac{1}{2}\right)^{k} \approx 2^{-k(1-h(s_0))}. \tag{8.5}$$

In this equation, h denotes the binary entropy function. Obviously, choosing a larger tag size will increase security against forging.

We will now give an upper bound for the probability of forging against a fixed participant. We compute the probability of passing at least one of the unknown $N-\omega$ tests, and use this to upper bound the probability that the coalition can forge a message sent to P_i. We find

$$\mathbb{P}(\text{FixedForge}) \leq 1 - (1 - p_t)^{N-\omega} \approx (N - \omega)p_t, \tag{8.6}$$

where we have used the fact that $p_t \ll 1$ in the approximation.

The total number of honest recipients is $N-\omega$ and for successful forging we only require that any one of them is deceived. Using the probability of forging against a fixed participant, we can bound the probability of deceiving any honest participant as

$$\mathbb{P}(\text{Forge}) = 1 - (1 - \mathbb{P}(\text{FixedForge}))^{N-\omega} \approx (N - \omega)^2 p_t, \tag{8.7}$$

where again we have used the fact that $\mathbb{P}(\text{FixedForge}) \ll 1$ in the approximation. Note that this probability decays exponentially fast with parameter k, and thus the protocol is secure against forging attempts. □

8.3.2 Transferability

Recall the definition of non-transferability, provided in Definition 4.10. In order to break the transferability of the protocol, a coalition C (which includes the signer P_0) must generate a signature that is accepted by recipient $P_i \notin C$ at level l, and rejected by another recipient $P_j \notin C$ at a level $l' < l$.

Theorem 8.3. *The protocol defined in Section 8.2 is secure against non-transferability attempts.*

Proof. The task of the coalition is easiest if $l' = l - 1$ and so we consider this case in what follows. To provide an upper bound on the cheating probability, we allow for the biggest coalition C, i.e. one that includes Nd_R recipients and the sender. For simplicity, we will again start by fixing the participants whom the coalition is trying to deceive to be the honest participants P_i and P_j. All other honest participants will be labelled with the index h. In general, transferability fails if the coalition forms a signature that is not transferable for at least one pair of any honest participants (P_i, P_j). Therefore, we should take into account all possible pairs of honest participants. We begin by focusing on the case of a fixed pair of participants, and at the end we give the more general expressions.

The first step is to compute $p_{m_{l,l-1}}$, which is the probability that: (i) test $T^m_{i,h,l}$ is passed (i.e. the tags sent from honest participant P_h to recipient P_i are accepted at level l); and (ii), the test $T^m_{j,h,l-1}$ fails (i.e. the tags sent from honest participant P_h to recipient P_j are rejected at level $l-1$). Since the sender P_0 is dishonest, it can be assumed that the coalition knows all of the signature functions. However, they are unaware of the sets $R_{h\to i}$ and $R_{h\to j}$. Therefore, the coalition can control the number of mismatches the signature will make with the signature functions originally sent to P_h, but they cannot separately bias the number of mismatches the signature will make with the functions in $F_{h\to i}$ and $F_{h\to j}$. Therefore, when participants P_i and P_j test the functions sent to them by an honest participant P_h, they will both have the same expected fraction of mismatches; we call this fraction p_e.

It is helpful to use the following bound

$$\begin{aligned} p_{m_{l,l-1}} &= \mathbb{P}(P_i \text{ passes test at level } l \wedge P_j \text{ fails test at level } l-1) \\ &\leq \min\{\mathbb{P}(P_i \text{ passes test at level } l),\ \mathbb{P}(P_j \text{ fails test at level } l-1)\}. \end{aligned} \tag{8.8}$$

The probability of passing the test at level l when $p_e > s_l$ can be bounded using

Hoeffding's inequalities to be below

$$\exp(-2(p_e - s_l)^2 k). \tag{8.9}$$

The probability of failing the test at level $l-1$ when $p_e < s_{l-1}$ can similarly be bounded to be smaller than

$$\exp(-2(s_{l-1} - p_e)^2 k). \tag{8.10}$$

Note that $s_{l-1} > s_l$ and so the above two cases cover all possible values for p_e. Therefore $p_{m_{l,l-1}}$ will always decay exponentially with the parameter k. As in [34], since we are taking the minimum over both cases, the optimal choice for the coalition is choose p_e so that the probabilities in Eqs. (8.9) and (8.9) are equal. This is achieved by choosing $p_e = (s_l + s_{l-1})/2$. In this case we obtain the bound

$$p_{m_{l,l-1}} \leq \exp\left(-\frac{(s_{l-1} - s_l)^2}{2} k\right), \tag{8.11}$$

which again decays exponentially with k.

For a test that involves a member of C it is trivial for the coalition to make two recipients disagree in any way they wish, i.e. they can make $T^m_{i,c,l}$ and $T^m_{j,c,l-1}$ take any values they wish if $P_c \in C$. However, the number of those tests is at most Nd_R, which is the maximum number of recipients in the coalition. For the participant P_i to accept a message at level l, he needs at least $N\delta_l + 1$ of the tests to pass at this level, as per Eq. (8.3). On the other hand, for the participant P_j to reject the message at level $l-1$, at most $N\delta_{l-1}$ of tests must pass at this level. Therefore, since it holds that $\delta_l - \delta_{l-1} = d_R$, in order for the coalition to be successful, the honest participants P_i and P_j need to disagree on at least $Nd_R + 1$ tests. As we saw, the coalition can easily make them disagree on the Nd_R tests originating from coalition members, but they still have to disagree on at least one more test originating from an honest recipient. There are $N(\delta_l - d_R) + 1$ such tests (tests originating from an honest recipient that were passed by P_i), and the P_j need only reject one of them for the coalition to succeed. Therefore, we can bound the probability of non-transferability between P_i and P_j by the probability that they disagree on a single test originating from an honest participant. We find

$$\begin{aligned} \mathbb{P}(\text{Fixed Non-Transferability}) &\leq 1 - (1 - p_{m_{l,l-1}})^{N(\delta_l - d_R)+1} \\ &\approx (N(\delta_l - d_R) + 1) p_{m_{l,l-1}}. \end{aligned} \tag{8.12}$$

Lastly, we consider the general case, where the participants P_i and P_j are not fixed. Simple combinatorial arguments give

$$\begin{aligned}\mathbb{P}(\text{Non-Transferability}) &\leq 1-(1-\mathbb{P}(\text{Fixed Non-Transferability}))^{N_p} \\ &\approx N_p(N(\delta_l-d_R)+1)p_{m_{l,l-1}},\end{aligned} \tag{8.13}$$

where $N_p := [(N(1-d_R)][N(1-d_R)-1]/2$. Again, this decays exponentially with k, and thus the protocol is secure against non-transferability. □

8.3.3 Repudiation

Recall the definition of repudiation, provided in Definition 4.11. Security against repudiation can be reduced to the special case of non-transferability from level $l = 0$ to level $l = -1$, thus we have the following:

Theorem 8.4. *The protocol defined in Section 8.2 is secure against repudiation attempts.*

Proof. The proof is a special case of non-transferability (see Section V A of [34]). We find

$$\mathbb{P}(\text{Repudiation}) \leq N_p(N(\delta_0-d_R)+1)p_{m_{0,-1}}. \tag{8.14}$$

This tends to zero exponentially fast with k, and thus the protocol is secure against repudiation. □

We note here that equations (8.7), (8.13) and (8.14) are independent of the message size, meaning the signature size will be constant with respect to the size of the message being sent.

8.4 Resource requirements

Theorem 5.1 in Chapter 5 states that $O(N^2)$ authenticated channels are always necessary to sign a message with unconditional security. In addition to this, all previous protocols (both quantum e.g. Refs. [1, 110], and classical e.g. Refs. [37, 40, 42]) have also required secret channels, and have used them to transmit $O(n)$ bits, where n is the bit-length of the message to be signed.

In the information-theoretic setting there is no physical difference between an authenticated classical channel and a secret classical channel, since both can be created using any channel capable of transmitting bits. As we saw in Chapter 3,

the only difference is the amount of secret shared key required. To authenticate an n-bit message, the sender and receiver must share $O(\log n)$ bits of secret key [31]. On the other hand, to send the message in secret the sender and receiver must share $O(n)$ bits of secret key [7].

Therefore, it is sometimes misleading to talk about resource requirements in terms of authenticated versus secret classical channels, as in the standard resource model. Instead, one should talk about the number of shared secret bits required, since this is what is used to create both authenticated and secret channels in the information-theoretic setting.

The hash scheme resource scalings

Importantly, the hash scheme uses secret key only to transmit the keys (hash functions) in secret (and to authenticate this communication). As per Theorem 8.1, the number of bits needed to specify a single key is logarithmic in the bit-size of the message being sent. This means the amount of secret shared key required by each participant in the hash scheme is $y = O(\log n + \log\log n)$, where the double log term comes from the need to authenticate the communication.

Accordingly, in terms of the scaling of resource requirements, using the hash scheme to sign an n-bit message m is no more expensive to implement than using a MAC scheme to authenticate message m. This protocol therefore provides the functionality of signatures at the same asymptotic cost as authentication. It could also be said that the hash scheme uses fewer resources than those assumed in the standard resource model, since the secret channels are only necessary to send very small messages, meaning their secret-bit cost is $O(\log n)$, rather than $O(n)$. This is important because it shows that the signing functionality is fundamentally (and significantly) cheaper than secrecy.

Dishonest participants

The number of dishonest participants the protocol is able to tolerate is directly related to the number of allowed transferability levels, according to the parameter $\delta_l = 1/2 + (l+1)d_R$. Specifically, the maximum transferability level for a given number of dishonest participants is set by the requirement that $\delta_l < 1$, meaning

$$(l_{\max} + 1)d_R < 1/2. \tag{8.15}$$

This limit is rather restrictive, and it is unclear whether this requirement is a result of our protocol, or whether it is a fundamental restriction on USS schemes using the resources in the standard resource model. We note that currently all quantum schemes suffer from the same restriction as above. However, it would be an interesting open question to see whether there are schemes which can tolerate more dishonest participants. The restriction seems to occur due to the exchange process carried out by all recipients.

In that case, it seems plausible that the same-state quantum USS schemes considered in Section 5.2, although highly inefficient, could tolerate higher numbers of dishonest participants than any classical USS protocol by avoiding the costly exchange process. Same-state protocols are not possible classically with unconditional security, because if all participants receive the same information it is always possible to forge. Increasing the allowable number of dishonest participants could therefore be a distinct advantage provided by quantum schemes. Nevertheless, further research is necessary to confirm this.

8.5 Protocol extensions

Reusability

A desirable extension of the current protocol would be to make the distributed keys reusable so that multiple messages could be signed using a single distribution stage. With the current protocol, this is not possible – the definition of an ϵ-ASU$_2$ set means that to maintain security against forging attempts, once the keys have been used to sign a message they must be discarded.

Reusability could be obtained in two different ways. The first (trivial) method would simply be to perform the distribution stage ψ times before moving on to the messaging stage. In this way, the sender would be able to send ψ different messages in the future. The second method would be to distribute functions from an ϵ-ASU$_\psi$ set, instead of an ϵ-ASU$_2$ set as described above.

Definition 8.5 ([149])**.** A hash function family $\mathcal{F}$ of functions from $\mathcal{M}$ to $\mathcal{T}$ is ϵ-ASU$_\psi$ provided that for all distinct elements $m_1, ..., m_\psi \in \mathcal{M}$ and for all (not necessarily distinct) $t_1, ..., t_\psi \in \mathcal{T}$, we have

$$|\{f \in \mathcal{F} : f(x_i) = y_i, 1 \leq i \leq \psi\}| \leq \epsilon \times |\{f \in \mathcal{F} : f(x_i) = y_i, 1 \leq i \leq \psi - 1\}|.$$

The meaning of this definition is that a function chosen randomly from $\mathcal{F}$ sim-

ulates a truly random function when up to $\psi - 1$ input-output pairs are known. It is shown in [149] that such families exist. Similarly to how ϵ-ASU$_2$ families allow us to sign a single message with unconditional security, ϵ-ASU$_\psi$ families allow us to sign $\psi - 1$ messages with unconditional security.

The smallest known ϵ-ASU$_\psi$ family requires $z = (\psi^2 - \psi + 1)b + \psi v$ bits to specify a function, where v is an integer such that $a \leq ((\psi - 1)b + v)(1 + 2^v)$, and where a and b are the message and tag length respectively, as before. Note that for $\psi = 2$ this reduces to the key length given in (8.1).

For simplicity, we will have the sender perform the distribution stage ψ times, rather than using an ϵ-ASU$_\psi$ family. Note that for a fixed tag length and large v (with $\psi \ll v$), both methods require the distribution of $O(\psi s)$ secret keys to leading order, where s is defined in Theorem 8.1. Therefore, in our case there is little advantage in using ϵ-ASU$_\psi$ functions as opposed to simply performing the distribution stage ψ times.

Latecomers

One might wonder whether it is possible for a new participant to enter the protocol after the distribution stage. In fact it is, but it requires either a trusted authority (see below), or for the new participant to communicate with all existing participants in the protocol. More concretely, to join, the sender would give the new participant $(N + 1)k$ functions from the ϵ-ASU$_2$ set. The participant would then send k of the functions to each of the other recipients and keep k for himself. The other participants would each randomly select k of the Nk functions they hold and send them over secure channels to the latecomer. Following this, security follows in a very similar manner as before.

Designated sender

For practical applications of signature schemes, it is often useful for any participant to be able to sign a message, rather than having a designated sender. This can trivially be introduced to the hash scheme by having the participants perform the distribution stage $N + 1$ times, where each participant acts as the sender in one of the distribution stages.

Trusted authority

The hash scheme requires participants to communicate pairwise with all other participants, as well as for secret keys to be distributed pairwise. For some applications, this may be too cumbersome a requirement, especially when all future participants are not known. In those situations, it is possible to greatly increase the efficiency of the protocol at the expense of introducing a trusted authority.

In the distribution stage, the signer would send Nk functions to the trusted authority, where N is an arbitrarily large number chosen to be the maximum number of participants able to verify the senders signature. When the sender wants to send a signed message, the trusted authority randomly (and secretly) sends k of the Nk functions to the recipient. Recipients could either obtain their k functions at the start of the protocol (i.e. have a distribution stage), or simply request the functions from the trusted authority as and when needed. Security against forging would follow as before from the properties of ϵ-ASU_2 sets, while security against repudiation would come from the fact that the trusted authority distributes the functions out at random, so each honest participant would have the same expected number of mismatches with any signature declaration. This would simplify the protocol in that all participants would only need to share a short secret key with the trusted authority, rather than requiring pairwise secret shared keys. Thus, as well as removing the need for pairwise communication between all parties, the total number of secret shared bits needed to generate the verification algorithms would scale as $O(N)$, rather than $O(N^2)$ as in the unmodified protocol.

Further benefits are that messages would be transferable an unlimited number of times between participants, and that if the sender gives an excess of keys to the trusted authority, latecomers can easily join by communicating solely with the trusted authority, who would send the latecomer k of the unused functions.

Extended protocol

In the section that follows, to facilitate comparisons to other classical USS schemes we include some of the above extensions into the basic protocol described in Section 8.2. Namely, each participant will perform the distribution stage ψ times in the role of the sender. Thus the distribution stage is performed $(N+1)\psi$ times before the messaging stage takes place. In this case, any participant would be able to send up to ψ messages in future. We will refer to this as the *extended protocol*. Note that we do not include a trusted authority.

8.6 Comparisons to existing schemes

8.6.1 Classical USS schemes

In this section we compare the performance of our extended protocol to the classical USS scheme proposed in Ref. [40] constructed using polynomials over a finite field. We will refer to this protocol as the HSZI scheme. The hash scheme enjoys a number of advantages when compared to the HSZI scheme. Namely,

1. We require fewer trust assumptions – the hash scheme does not require a trusted authority.

2. Security in the hash scheme can be tuned independently of message size, resulting in shorter signature lengths.

3. The hash scheme scales more efficiently (with respect to message size) in terms of the number of secret shared bits required by participants.

We will look at the second and third advantages in more detail.

Signature length

According to Theorem 3 of [40] (translated to our notation) the HSZI scheme has

$$|\Sigma| = q^{(\omega+1)}, \tag{8.16}$$

$$|\mathcal{S}| = q^{(\omega+1)(\psi+1)}, \tag{8.17}$$

$$|\mathcal{V}| = q^{\omega+(N+1)(\psi+1)}, \tag{8.18}$$

where Σ is the set containing all possible signatures, $\mathcal{S}$ is the set containing all possible signing algorithms, $\mathcal{V}$ is the set containing all possible verification algorithms, q is the number of elements in the chosen finite field and ψ is the number of times the keys can be reused.

Let us first consider the size of the signature. Since the signature must be transmitted with the message, it is desirable to have as small a signature as possible. In the HSZI scheme the message m is an element of the finite field, meaning the size of the finite field must be at least as big as the size of the message set, i.e. $q \geq |\mathcal{M}|$. Accordingly, in what follows we set $q = |\mathcal{M}|$. Eq. (8.16) implies that $(\omega + 1)\log(|\mathcal{M}|)$ is the bit-length of the signature. The authors also show that the HSZI scheme provides security proportional to $1/|\mathcal{M}|$.

Immediately then, we see that both the size of the signature and the security level depend on the size of the message to be sent. On the other hand, in the hash scheme, the signature length is $2N^2k$ bits, regardless of the message length. The security level in the hash scheme depends on the parameter k, but is independent of the length of the message being signed. This allows the hash scheme to bypass the optimality results presented in Ref. [40]. Specifically, the authors show that the signature generated by the HSZI scheme is optimally small *for a given security level*. By decoupling the security level from the size of the message being sent, we are able to generate smaller signatures while maintaining security.

Secret key requirements

We now consider the number of secret shared bits required to securely distribute the signing/verification keys. In the HSZI scheme, to secretly send the signing and verification keys to all participants, the trusted authority must hold

$$\big[(\omega+1)(\psi+1)+\omega+(N+1)(\psi+1)\big]\log(|\mathcal{M}|) = O(N\psi\log|\mathcal{M}|) \tag{8.19}$$

secret shared bits with each participant (as implied by Eqs. (8.17) and (8.18)).

For the hash scheme, each recipient must share Nky secret bits with the sender (to receive the signature functions), and $ky+k\log(Nk)$ with every other recipient (to forward on a selection of the key functions and their positions). For the extended protocol, where the distribution stage is performed ψ times for each participant acting as sender, each participant must share: (i) Nky secret bits with each of the N recipients for the ψ rounds in which he is the sender; and (ii) Nky bits with the sender and $ky+k\log(Nk)$) secret bits with each of the $(N-1)$ other recipients for each of the $N\psi$ rounds when he is not the sender. This is a total of

$$\begin{aligned} N^2k\psi y + N\psi\big[Nky + k(N-1)(y+\log(Nk))\big] & \\ &= Nk\psi(3N-1)y + N(N-1)k\psi\log(Nk) \\ &= Nk\psi(3N-1)(6+2s) + N(N-1)k\psi\log(Nk) \\ &= O\big(N^2k\psi(\log\log|\mathcal{M}| + \log Nk)\big) \end{aligned} \tag{8.20}$$

secret shared bits per recipient. The second equality follows using the definition of y together with $b=2$. The last equality follows using the Lambert W function to find a leading order approximation for s when s is large [150]. The results are summarised in Table 8.1 below.

The table shows that the signature length in the hash scheme is constant with respect to the size of the message to be signed. On the other hand, the signature length in the HSZI scheme increases linearly with the bit-length of the message to be signed. Similarly, the secret shared key required by the hash scheme increases logarithmically with the bit-length of the message, whereas the increase in the HSZI scheme is linear in the bit-length of the message.

The fact that the hash scheme scales unfavourably with respect to the number of participants is due to the lack of a trusted authority, meaning participants must perform the pairwise exchange process. As mentioned in Section 8.5, this N^2 scaling can be removed from the hash scheme by introducing a trusted authority.

	Hash scheme	HSZI	Quantum USS
Signature size	$2N^2k$	$(\omega+1)a$	$O(N^2a)$
Secret key size	$O\big(N^2\psi(\log a+\log N)\big)$	$O(N\psi a)$	$O\big(N^2\psi(a+\log N)\big)$

Table 8.1: Comparison of the signature length and secret shared key required for various signature protocols. It can be seen that the hash scheme scales favourably with respect to the message length, $a := \log|\mathcal{M}|$, both in terms of signature length and required secret shared key. The "Quantum USS" column refers to practical quantum USS schemes in general. Though there are many such schemes, the above rates are applicable to the schemes which at present are most efficient, namely, the AWKA protocol and GP2.

Disadvantages

Due to the inclusion of a trusted authority, the HSZI scheme enjoys a number of advantages over the hash scheme. These are:

1. Pairwise secret shared keys between all participants are not required by the HSZI scheme. Instead, each participant only needs a shared secret key with the trusted authority. This means that the HSZI scheme scales favourably with respect to the number of protocol participants.

2. Participants in the HSZI scheme are able to enter the protocol even after the distribution stage. The new participant only needs to communicate with the trusted authority to join.

3. The HSZI protocol has unlimited transferability, whereas the hash scheme can only guarantee transferability a finite number of times.

While these advantages are significant, they are only possible due to the existence of a trusted authority – an additional trust assumption not present in the hash scheme. As highlighted in Section 8.5 the hash scheme could easily be modified to include

the trusted authority, in which case it would achieve the same three benefits above, as well as being significantly more efficient.

8.6.2 Quantum USS schemes

A central motivating factor in the study of quantum USS schemes was that they usually require fewer resources than classical USS schemes. Unsurprisingly, this benefit came at a cost, and all quantum USS schemes proposed have been much less efficient than classical USS schemes in terms of signature length and signature generation times[3].

Until now, this decrease in efficiency had been partly justified by the fact that quantum protocols do not require detectable broadcast channels, anonymous channels, or a trusted authority. Instead, the only assumptions are that a limited number of the participants are dishonest, and that the participants all share a number of secret bits, which could be expanded via QKD.

However, the hash scheme makes *the same* trust assumptions as quantum USS schemes, and still achieves two key advantages. Namely, the hash scheme generates much shorter signatures and requires significantly fewer secret shared bits. One of the reasons for the increase in efficiency is that, so far, all quantum USS schemes have been of the Lamport-type, in which the distribution stage must be performed for every possible future message. On the other hand, the hash scheme does not follow this blueprint, and instead requires users to share hash functions in the distribution stage, which can be used to sign any future message (up to some chosen size).

Efficiency

Here we consider the signature length and secret shared bit requirements of the hash scheme and compare it to GP2. Although GP2 is essentially a classical USS scheme (and should be classified as such in this author's opinion), it was originally described with the assumption that all participants generate and distribute secret shared key using QKD. As such, the authors presented it as a quantum USS scheme and, if classified as such, it is still the *only* quantum USS scheme for which a full N-party security analysis exists. It would also be the most efficient quantum USS scheme that is experimentally realisable, meaning the efficiency benefits of the hash scheme (described in this subsection) apply equally to more distinctly "quantum"

[3]Although it may appear from Table 8.1 that quantum USS schemes scale comparably to the HSZI scheme, in fact the constant of proportionality for the quantum schemes is very large, meaning that for all practical purposes the HSZI scheme is far more efficient.

USS schemes (such as the AWKA scheme). For these reasons, we have chosen to compare the hash scheme to GP2.

We assume that a group of $N + 1 = 51$ participants are trying to sign a 1 Mb message to a security level of 10^{-10}. To make the comparison to GP2 fair, rather than considering the extended protocol, we assume that the participants perform the distribution stage as specified in Section 8.2, i.e. there is a designated sender and only one message to be sent. In order to have $l_{\max} = 1$, we assume that at most $\omega = 13$ participants are dishonest meaning $d_R = 0.24$. We also choose $s_{-1} = 0.41$, $s_0 = 0.21$ and $s_1 = 0.01$ so as to have even gaps between the verification levels[4].

With these parameters, Eqs. (8.7), (8.13) and (8.14) show that $k = 1700$ is necessary for the message to be secure to a level of 10^{-10}. Given this value of k, the signature length is 8.50×10^6 and each recipient must hold a total of 7.69×10^6 secret shared bits (shared over the different participants).

When considering GP2, we assume the sender signs the 1 Mb message bit-by-bit, each to a level of 10^{-10}. Overall this gives a lower security level than signing the message as a whole, but makes the protocol significantly more efficient[5]. Eqs. (24), (29) and (31) of Ref. [34] can be used to show that the resulting signature length is 4.25×10^{12}, and that each recipient must hold a total of 5.96×10^{12} secret shared bits (shared over the different participants).

This example shows just how powerful the hash function scheme is when compared to quantum schemes – even for a relatively small message, the hash scheme is 6 orders of magnitude more efficient both in terms of signature size and resource requirements. Our results show that quantum USS schemes must either be drastically improved, or find a new source of motivation if they are to remain relevant.

8.6.3 Computationally secure digital signatures

In this section we compare the hash scheme to some of the most popular computationally secure digital signature schemes. The comparison is fraught with difficulties since, in many respects, USS schemes are fundamentally different to digital signatures. Nevertheless, we think the comparison is worth a try.

In Table 8.2 we state the signature length as well as the public and private key sizes for various common digital signature schemes. For comparison, Table 8.3 gives the secret key requirements and signature length of the hash scheme for the same

[4]This choice is somewhat arbitrary, but is chosen to minimise the required signature lengths.

[5]Signing the message as a whole would require participants to share secret keys of size $O(|\mathcal{M}|) = O(2^{10^6})$, which is clearly impossible.

security level.

Algorithm	Public key	Private key	Signature size
RSA [12]	3,072	24,576	3,072
DSA [13]	3,072	3,328	3,072
ECDSA [14]	512	768	512
XMSS (Hash based) [27]	7,296	152	19,608
Bliss (Lattice based) [19]	7,000	2,000	5,600
Rainbow (Multivariate) [22]	842,400	561,352	264

Table 8.2: This table shows the public key length, private key length, and signature size of various common digital signature schemes [26]. The schemes on rows 1-3 are computationally secure in the classical setting but not quantum-safe. The schemes on rows 4-6 are quantum-safe. The figures are quoted in bits, and are the lengths required for 128-bit security, i.e. a security level of 2^{-128}.

Algorithm	Secret shared key	Signature size
Hash scheme	45,250,100	43,500,000
Trusted Authority	95,200	220,000

Table 8.3: This table shows the secret key requirements (per participant) and signature size needed to sign a single 1 Mb message between 51 participants with 128-bit security using the hash scheme. The figures are quoted in bits. The first row is for the protocol as described in Section 8.2, while the second row allows for a trusted authority as described in Section 8.5.

Recall that digital signatures are *believed* to provide computational security, rather than the unconditional security provided by USS schemes. The top three lines of Table 8.2 show the schemes that are most commonly used in the real world. These schemes are not quantum-safe, i.e. in the presence of quantum adversaries the schemes are proven to be completely insecure [15]. The bottom three lines of Table 8.2 show the most likely successors to the current digital signature schemes. These schemes are believed to be quantum-safe, which means they are believed to provide computational security even in the presence of quantum adversaries. As a consequence of the lower security level provided, digital signatures also enjoy some additional advantages not explicitly stated in the tables above. Namely,

1. Digital signatures are public-key schemes and do not require any secret shared key between participants.

2. Digital signatures are universally verifiable.

3. The signature length and public/private-key sizes do not depend on the number of participants in the scheme.

4. Public and private keys can be reused to sign many messages[6].

Clearly, the tables and the points above show that the hash scheme is still less efficient than the competing quantum-safe digital signature schemes, though the difference is perhaps not as large as expected, particularly if one allows for a trusted authority.

Nevertheless, even without the trusted authority, the hash scheme requires participants to share a total of 4.35×10^7 secret bits (spread across the other participants) in order to send/receive a 1 Mb message with 128-bit security, or 7.69×10^6 if the security level is reduced to 10^{-10}. While this might sound like a lot, it is worth noting that standard QKD systems can already distribute secret key at a rate in excess of 1 Mbps [151] and this rate is constantly increasing. As such, the hash scheme can certainly be considered practical and within the reach of current technology.

A potential advantage of the hash scheme is the computational efficiency of generating the signatures and the verification keys. To varying degrees, all of the digital signature schemes above are quite computationally intensive when it comes to creating a signature. In many applications this is not an issue, but for settings where there are limited computational resources available, creating the signatures may cause a noticeable slowdown of the application. The hash scheme on the other hand requires only the evaluation of universal hash functions, something which is often computationally cheap. For example, many commonly used ϵ-ASU_2 sets are created from Toeplitz matrices (e.g. [99, 152]) whose evaluation is simple and efficient.

It should be stressed that in real-world applications, the requirement of shared secret keys mean that USS schemes should not be considered a stand-alone product. Rather, they should be thought of as a complement to existing QKD networks. Clearly, any system connected via a network of QKD links values high security. In this case, the additional security guarantees offered by USS schemes over digital signatures may be a significant incentive for their use. Further, the implementation of USS schemes in existing QKD networks would come at a very small additional cost, since the infrastructure necessary to distribute secret keys would already be in place.

[6]There are limits on how many messages can be signed using XMSS, but the number is very large $\approx O(2^{20})$.

8.7 Conclusion

In this chapter we introduced a classical USS scheme which required fewer resources than all previous classical USS schemes proposed in the literature – namely, we presented a secure scheme that did not rely on either a trusted authority, broadcast channels or anonymous channels. Further, to sign an n-bit message, the hash scheme used secret channels only to send communications $O(\log n)$ in size, as opposed to $O(n)$ as is necessary in P2 [1], GP2 [34] and all known quantum USS schemes. As such, our scheme has smaller resource requirements than all known quantum USS schemes. Despite this, we show that in comparison to all quantum USS schemes, the hash scheme is far superior, achieving efficiency improvements of at least six orders of magnitude. As such, it is unclear what advantages quantum USS schemes may provide over classical USS schemes, and additional motivation is necessary if further quantum schemes are proposed.

In comparison to existing classical USS schemes, the hash scheme is again more efficient both in terms of the signature length and the secret key requirements. In fact, it is shown that the cost of implementing the hash scheme scales in the same way as message authentication, and the hash scheme can therefore be considered cheap.

Lastly, we compared the hash scheme to a selection of some of the most common public-key digital signature schemes, both quantum-safe and not. We found that, overall, the efficiency shortcomings of USS schemes mean they are certainly not going to replace quantum-safe digital signatures in most real world applications. Nevertheless, the hash scheme can be considered practical with current technology, and can be implemented within existing QKD networks for a low additional cost. Therefore, for systems requiring very high levels of security, the hash scheme could well find commercial application.

Chapter 9

Imperfect oblivious transfer

9.1 Introduction

The results of the previous chapter show that classical USS schemes can be drastically more efficient than all known quantum USS schemes. Importantly, this is the case even for classical schemes requiring the same (or fewer) resources than quantum schemes. As such, it is unclear whether quantum mechanics is necessary or useful in creating USS schemes.

One potential advantage of quantum schemes, highlighted in Section 8.4, is that the same-state schemes described in Section 5.2 may be able to increase the maximum tolerable number of dishonest participants within a USS protocol. In these same-state schemes, we require the guarantees that:

1. The recipient cannot gain full information on the states Alice sends (to protect against forging), and;

2. Alice does not know what information the recipient receives (to protect against repudiation/non-transferability).

As discussed in Section 3.7, these guarantees are highly reminiscent of 1-out-of-2 oblivious transfer (1-2 OT).

OT is one of the most important primitives in cryptography. Its importance stems from the fact that it can be used as the foundation for all secure two-party computations – with OT, all secure two-party computations are possible [102, 103]. The widespread use and applicability of OT means that, aside from its potential relevance to USS schemes, studying what is achievable with information-theoretic security is independently interesting, and the bounds that we prove may impact a wide range of other cryptographic protocols. The work in this chapter is taken from

Ref. [153] with minor modifications.

9.2 Background and related work

OT exists in many different flavours, all with slightly different definitions and notions of security. OT was first introduced informally in 1970 by Wiesner as "a means for transmitting two messages either but not both of which may be received" [154], and subsequently formalised as 1-2 OT in [155]. In related work, Rabin [156] introduced a protocol (now called Rabin OT), which was later shown by Crépeau [157] to be classically equivalent to 1-2 OT, in the sense that if it is possible to do one, it is possible to use this to implement the other. Various "weaker" variants of OT have also been proposed, most notably Generalised OT, XOR OT and Universal OT [158], but all have been shown to be equivalent to 1-2 OT [159] in the classical setting. The equivalence is believed to also hold in the quantum setting, but the reduction proofs may need to be revised. There is also work by Damgård, Fehr, Salvail and Schaffner [160] who define yet another variant of OT and characterise security in terms of information leakage. With these definitions (and their quantum counterparts), the authors describe a 1-2 OT protocol which is perfectly secure in the bounded quantum storage security model.

Following the discovery of quantum key distribution in 1984 [10], there arose a general optimism that quantum mechanics may provide a means to perform multiparty computations with information-theoretic security. Despite this early confidence, the history of secure two-party computations is characterised by mainly negative results. Mayers and Lo [104, 105] proved that all one-sided two-party computations are insecure in the quantum setting, meaning that it is impossible to perform important protocols such as bit commitment and OT with information-theoretic security. Nevertheless, the result does not exclude imperfect variants of these protocols from being possible, and it has been an interesting and productive open question to determine the optimal security parameters achievable for some important two-party computations.

For most cryptographic primitives, this question has been definitively answered. For strong coin flipping, Kitaev [161] introduced the semi-definite programming formalism to show that the product of Alice's and Bob's cheating probabilities must be greater than $1/2$, implying that the minimum cheating probability is at least $1/\sqrt{2}$. For weak coin flipping, Mochon [162] showed that the minimum cheating probability is at least $1/2$. In the same paper a protocol achieving this bound is

presented, showing that the bound is tight. Chailloux and Kerenidis [163] used these results on weak coin flipping to generate a protocol for strong coin flipping achieving Kitaev's bound. Lastly, for quantum bit commitment, Chailloux and Kerenidis [164] proved that the minimum cheating probability is 0.739, and presented a protocol achieving this bias. Thus, for bit commitment, weak coin flipping and strong coin flipping the achievability bounds are tight with the known protocols.

For OT on the other hand, the situation is not so clear-cut. The cheating probability of a 1-2 OT protocol is defined as $p_C := \max\{A_{OT}, B_{OT}\}$, where A_{OT} and B_{OT} are Alice's and Bob's ability to cheat, respectively (formal definitions of A_{OT} and B_{OT} are provided in Section 9.3). Classically, it is impossible to achieve even limited security for OT in the information-theoretic setting, since one party can always cheat with certainty so that $p_C = 1$. On the other hand, quantum mechanics allows for imperfect protocols, in which the participants are able cheat but their abilities are limited, i.e. $1/2 < p_C < 1$.

Contributions

In this chapter we consider stand-alone quantum protocols for 1-2 OT, and are concerned only with information-theoretic security. As mentioned above, perfect security (i.e. $p_C = 1/2$) in this setting is impossible. However, the no-go results of Mayers and Lo do not exclude imperfect variants of OT from being possible, and these variants may be useful in constructing other cryptographic primitives. For example, perfect 1-2 OT is not necessary to construct USS schemes, since we only require that Bob does not gain *full* information on the states sent by Alice. Therefore, as long as there exists an OT scheme in which Alice's and Bob's cheating probabilities are sufficiently restricted, we may be able to use that imperfect OT scheme to create a fully secure USS scheme.

The highest known lower bound on p_C in all 1-2 OT protocols is due to Chailloux, Gutoski and Sikora [106], who show that $p_C \geq 2/3$. However, known 1-2 OT protocols all have a cheating probability of at least $p_C = 0.75$. Therefore, there is a gap between what is known to be achievable, and what is known to be impossible. This chapter contains three main contributions:

1. We introduce the concept of Semi-random OT (Section 9.3) and prove an equivalence between cheating in 1-2 OT and Semi-random OT (Section 9.4).

2. In Section 9.5 we describe a general framework for Semi-random OT, and use it to increase the lower bound on p_C for 1-2 OT to 0.749 if the states in

the final round of the (honest) protocol are pure and symmetric. Our results also reproduce the known $p_C \geq 2/3$ bound in the completely general setting[1]. Additionally, our construction parametrises Alice's and Bob's ability to cheat in terms of a single quantity, F, related to the fidelity of the protocol output states. This parametrisation suggests how to construct schemes when one of either sender or receiver dishonesty is prioritised, and also allows us to derive new bounds for these settings. Such a scenario arises in the context of USS schemes [1, 110], and the derived bounds prove useful for understanding the potential application of imperfect OT to signatures.

3. Lastly, in Section 9.6 we illustrate our results by describing a protocol which relies on unambiguous state elimination (USE) measurements, and can be used to implement many runs of OT. The protocol serves to highlight the interesting connection between USE measurements and 1-2 OT, and provides a new application for this relatively underused type of measurement. The average security parameters achieved are almost equal to the bounds proved in this chapter, and the optimal cheating strategies are exactly those considered in the general framework in the preceding sections.

9.3 Definitions

Intuitively, 1-2 OT is a two-party protocol in which Alice chooses two input bits, x_0 and x_1, and Bob chooses a single input bit b. The protocol outputs x_b to Bob with the guarantees that Alice does not know b, and that Bob does not know x_{1-b}. A cheating Alice aims to find the value of b, while a cheating Bob aims to correctly guess both x_0, x_1.

Definition 9.1 (1-2 OT [166])**.** A 1-2 quantum OT protocol is a protocol between two parties, Alice and Bob, such that

- Alice has inputs $x_0, x_1 \in \{0,1\}$ and Bob has input $b \in \{0,1\}$. At the beginning of the protocol, Alice has no information about b and Bob has no information about (x_0, x_1).

- At the end of the protocol, Bob outputs y or Abort and Alice can either Abort or not.

[1]At the time of writing the results contained in this chapter, we believed the known lower bound to be $p_C \geq 0.585\ldots$ as per Ref. [165]. However, following the submission of our results we discovered existing work (Ref. [106]), performed independently to our own, which indirectly implies the increased lower bound of $p_C \geq 2/3$.

- If Alice and Bob are honest, they never Abort, $y = x_b$, Alice has no information about b and Bob has no information about x_{1-b}.
- $A_{OT} := \sup\{Pr[\text{Alice guesses } b \wedge \text{Bob does not Abort}]\} = \frac{1}{2} + \epsilon_A$.
- $B_{OT} := \sup\{Pr[\text{Bob guesses } (x_0, x_1) \wedge \text{ Alice does not Abort}]\} = \frac{1}{2} + \epsilon_B$.

The suprema are taken over all cheating strategies available to Alice and Bob. This definition of security against Bob differs from some other works, for example [167], in which security is characterised in terms of the information leakage, or in terms of Bob's ability to guess the output of some function $f(x_0, x_1)$, commonly the XOR. Nevertheless, our simpler definition makes sense if we are interested only in lower bounds on p_C, since the ability to guess (x_0, x_1) automatically implies the ability to guess $f(x_0, x_1)$ for any f. In other situations, the choice of which definition is most appropriate will be largely application dependent.

To prove the results contained in this chapter, we also introduce a useful variant of OT, which we call Semi-random OT. Semi-random OT differs from 1-2 OT in that Bob does not have any inputs and is randomly assigned an output.

Definition 9.2 (Semi-random OT)**.** 1-2 quantum Semi-random OT, or simply Semi-random OT, is a protocol between two parties, Alice and Bob, such that

- Alice chooses two input bits $(x_0, x_1) \in \{0, 1\}$ or Abort.
- Bob outputs two bits (c, y) or Abort.
- If Alice and Bob are honest, they never Abort, $y = x_c$, Alice has no information about c and Bob has no information on x_{1-c}. Further, if Alice and Bob are honest, c is a uniformly random bit.
- $A_{OT} := \sup\{Pr[\text{Alice guesses } c \wedge \text{Bob does not Abort}]\} = \frac{1}{2} + \epsilon_A$.
- $B_{OT} := \sup\{Pr[\text{Bob guesses } (x_0, x_1) \wedge \text{ Alice does not Abort}]\} = \frac{1}{2} + \epsilon_B$.

The reason for introducing Semi-random OT is that we have found it simpler to work with than 1-2 OT, and the ability to perform Semi-random OT with cheating probabilities A_{OT} and B_{OT} is equivalent to being able to perform 1-2 quantum OT with the same cheating probabilities (see Section 9.4). Therefore, the lower bounds on p_C that we prove for Semi-random OT also apply to the well known 1-2 OT.

9.4 Equivalence of Semi-random OT and 1-2 OT

In this section we prove the following equivalence between the cheating probabilities in Semi-random OT and 1-2 OT.

Proposition 9.3. *The existence of a Semi-random OT protocol with cheating probabilities A_{OT} and B_{OT} is equivalent to the existence of a 1-2 OT protocol with the same cheating probabilities.*

To prove this, we begin by introducing a related OT variant called Random OT (ROT). ROT differs from Semi-random OT in that Alice has no inputs, and is instead randomly given two outputs.

Definition 9.4 (Random OT)**.** Random OT is a protocol between two parties, Alice and Bob, such that

- Alice outputs two bits $(x_0, x_1) \in \{0, 1\}$ or Abort.
- Bob outputs two bits (c, y) or Abort.
- If Alice and Bob are honest, they never Abort, $y = x_c$, Alice has no information about c and Bob has no information on x_{1-c}. Further, if Alice and Bob are honest, x_0, x_1 and c are uniformly random bits.
- $A_{OT} := \sup\{Pr[\text{Alice guesses } c \wedge \text{Bob does not Abort}]\} = \frac{1}{2} + \epsilon_A$.
- $B_{OT} := \sup\{Pr[\text{Bob guesses } (x_0, x_1) \wedge \text{Alice does not Abort}]\} = \frac{1}{2} + \epsilon_B$.

Ref. [166] proved that the existence of a ROT protocol with cheating probabilities A_{OT} and B_{OT} is equivalent to the existence of a 1-2 OT protocol with the same cheating probabilities. Following very similar arguments, in the following subsections we will show that the existence of a Semi-random OT protocol with cheating probabilities A_{OT} and B_{OT} is equivalent to the existence of a ROT with the same cheating probabilities. This, combined with the results in Ref. [166], proves the proposition.

Semi-random OT from ROT

Let P be a ROT protocol with cheating probabilities $A_{OT}(P)$ and $B_{OT}(P)$. We construct a Semi-random OT protocol with the same cheating probabilities as follows:

1. Alice has inputs (z_0, z_1).

2. Alice and Bob run protocol P to output (x_0, x_1) for Alice and (c, y) for Bob.

3. Alice and Bob abort in Q if and only if they abort in P. Otherwise, Alice sends $(z_0 \oplus x_0, z_1 \oplus x_1)$ to Bob.

4. Bob outputs (c, y') where $y' = (z_c \oplus x_c \oplus y)$.

We now show that Q is a Semi-random OT protocol with cheating probabilities $A_{OT}(P)$ and $B_{OT}(P)$.

If Alice and Bob are honest, then by definition we have $y = x_c$ and so $y' = z_c$. Alice has no information on c and Bob has no information on z_{1-c}. Further, $c \in_R \{0, 1\}$ as required.

If Alice is dishonest, she cannot guess c except with probability $A_{OT}(P)$ since she only receives communications from Bob via protocol P. Therefore $A_{OT}(Q) = A_{OT}(P)$.

If Bob is dishonest, he holds $(z_0 \oplus x_0, z_1 \oplus x_1)$ and aims to guess (z_0, z_1). This is equivalent to Bob guessing (x_0, x_1) which he can do with probability $B_{OT}(P)$. Therefore $B_{OT}(Q) = B_{OT}(P)$.

ROT from Semi-random OT

Let P be a Semi-random OT protocol with cheating probabilities $A_{OT}(P)$ and $B_{OT}(P)$. We construct a ROT protocol Q with the same cheating probabilities as follows:

1. Alice picks $x_0, x_1 \in_R \{0, 1\}$ uniformly at random.

2. Alice and Bob perform the Semi-random OT protocol P where Alice inputs x_0, x_1. Let (c, y) be Bob's outputs.

3. Alice and Bob abort in Q if and only if they abort in P. Otherwise, the outputs of protocol Q are (x_0, x_1) for Alice and (c, y) for Bob.

The outputs of Q are uniformly random bits (in the honest case) since Alice chooses her input at random. Note that, in the definition of ROT, the outputs are only required to be random in the honest case, and no assertions are made when one party acts dishonestly. Therefore Q does indeed implement ROT. From the construction of Q it is also clear that $A_{OT}(Q) = A_{OT}(P)$ and $B_{OT}(Q) = B_{OT}(P)$. This concludes the proof of Proposition 9.3.

9.5 Generic protocol

In this section we introduce a general framework for Semi-random OT and use it to prove lower bounds on p_C for any Semi-random OT protocol. To do this, we present undetectable cheating strategies always available to Alice and Bob and analyse them to lower bound their cheating probabilities, A_{OT} and B_{OT} respectively. We show that for any Semi-random OT protocol

$$p_C = \max\{A_{OT}, B_{OT}\} \geq 2/3. \tag{9.1}$$

Further, if the possible states output to Bob by the (honest) protocol are pure and symmetric, then

$$p_C = \max\{A_{OT}, B_{OT}\} \geq 0.749. \tag{9.2}$$

We note that all 1-2 OT protocols we have seen proposed have output states that are pure and symmetric. Although there is no reason why this must be the case in general, the inherent symmetry of the protocol seems to lead to this property.

We will prove the above bounds by expressing Alice's and Bob's cheating probabilities in terms of a single parameter, F, related to the fidelity of the output states of the protocol. From this we find that there is always a trade-off; as Alice's ability to cheat decreases, Bob's ability increases, and vice versa.

9.5.1 Protocol framework

We now describe the general framework for Semi-random OT protocols with N rounds of communication between Alice and Bob. The framework is based on Kitaev's construction for strong coin flipping [161] and is useful for analysing the security of Semi-random OT.

1. Bob starts with the state ρ_{BM} and Alice starts with an auxiliary system A initialised to $|0\rangle\langle 0|_A$. The overall state is $\rho_{BMA} := \rho_{BM} \otimes |0\rangle\langle 0|_A$. We further suppose Alice and Bob share the counter variable i, initialised to 1, which tracks the round number of the protocol.

2. Alice randomly selects an element $x_0x_1 \in \{00, 01, 11, 10\}$.

3. Bob sends system M to Alice.

4. Based on her choice in Step 2, Alice performs the unitary operation $U_{MA}^{x_0x_1,i} \in \{U_{AM}^{00,i}, U_{AM}^{01,i}, U_{AM}^{11,i}, U_{AM}^{10,i}\}$.

5. Alice sends system M back to Bob.

6. Bob performs the unitary operation $V_{BM}^{(i)}$.

7. The index i is incremented by 1. If $i = N + 1$, the protocol proceeds to Step 8, otherwise it returns to Step 3.

8. The final output held by Bob is

$$\sigma_{BM}^{x_0x_1} := \mathrm{Tr}_A(\eta_{BMA}^{x_0x_1}), \tag{9.3}$$

where

$$\eta_{BMA}^{x_0x_1} := \mathcal{U}^{x_0x_1}\rho_{BMA}\left(\mathcal{U}^{x_0x_1}\right)^\dagger \tag{9.4}$$

and

$$\mathcal{U}^{x_0x_1} = V_{BM}^{(N)}U_{MA}^{x_0x_1,N}\ldots V_{BM}^{(1)}U_{MA}^{x_0x_1,1}. \tag{9.5}$$

9. Bob performs a POVM with elements $\{\Pi_{BM}^{0*}, \Pi_{BM}^{1*}, \Pi_{BM}^{*0}, \Pi_{BM}^{*1}\}$ to obtain the value of c and x_c. For example, the outcome Π_{BM}^{1*} denotes that $c = 0$ and $x_0 = 1$.

The steps of the framework above describes the honest actions of Alice and Bob, together with the associated outputs, assuming all measurements are deferred. Of course, Alice's and Bob's actual actions may deviate from the honest protocol description if they are dishonest.

In order to prove lower bounds on the protocol cheating probability p_C, in the following sections we will describe general cheating strategies that are always available to Alice and Bob within this framework, and which will always be undetectable.

9.5.2 Honest case

For the protocol to be correct in the honest case, we require the following conditions to hold:

$$\text{For } c = 0\text{:}\quad \mathrm{Tr}(\Pi_{BM}^{j*}\sigma_{BM}^{kl}) = \begin{cases} 1/2, & \text{if } j = k, \\ 0, & \text{if } j \neq k. \end{cases} \tag{9.6}$$

$$\text{For } c = 1\text{:}\quad \mathrm{Tr}(\Pi_{BM}^{*j}\sigma_{BM}^{kl}) = \begin{cases} 1/2, & \text{if } j = l, \\ 0, & \text{if } j \neq l. \end{cases} \tag{9.7}$$

These conditions imply that Bob receives either one of Alice's two chosen bits with equal probability, and that the bit received by Bob is correct.

9.5.3 Security against Bob

If Bob acts honestly throughout the protocol up until step 9, then at the beginning of this step he holds either σ_{BM}^{00}, σ_{BM}^{01}, σ_{BM}^{11}, or σ_{BM}^{10}. In order to cheat, Bob wants to guess the exact value of x_0 and x_1. Equivalently, Bob wants to know exactly which of the four σ states he holds.

To do this, Bob's optimal measurement is a minimum-error measurement. However, the minimum-error measurement will vary according to the states chosen by any specific implementation of Semi-random OT. Instead, to provide a lower bound on Bob's optimal cheating probability for *all* protocols, we assume that Bob performs the Square Root Measurement (SRM) [168]. Again, this may not be his optimal strategy, but it is a valid cheating strategy for any Semi-random OT protocol, and one that he can employ without risk of being caught (since there is no further interaction between Alice and Bob after step 7, so Alice has no way of knowing which measurement Bob performs). Using the success probability of the SRM, we can bound Bob's optimal cheating probability as [169]

$$B_{OT} \geq 1 - \frac{1}{8} \sum_{jk \neq lm} F(\sigma_{BM}^{jk}, \sigma_{BM}^{lm}), \tag{9.8}$$

where $jk, lm \in \{00, 01, 11, 10\}$ and F is the fidelity, defined as

$$F(\rho, \sigma) := \mathrm{Tr}\left(\sqrt{\rho^{1/2}\sigma\rho^{1/2}}\right). \tag{9.9}$$

Eqs. (9.6) and (9.7) imply that $F(\sigma_{BM}^{jk}, \sigma_{BM}^{j\oplus 1, k\oplus 1}) = 0$ (since these states can be perfectly distinguished). Without loss of generality, we suppose σ_{BM}^{00} and σ_{BM}^{01} are the pair with the highest fidelity. Define

$$F := F(\sigma_{BM}^{00}, \sigma_{BM}^{01}). \tag{9.10}$$

Then

$$B_{OT} \geq 1 - F. \tag{9.11}$$

This result is limited somewhat by the bound on the success probability of the SRM for general mixed states, given in Eq. (9.8). Placing restrictions on the output states of the protocol allows us to tighten this bound. In particular, if $\{\sigma_{BM}^{00}, \sigma_{BM}^{01}, \sigma_{BM}^{11}, \sigma_{BM}^{10}\}$ forms a symmetric set[2] of pure states, then Bob's is successful in guessing

[2]Symmetric sets of states are ubiquitous in quantum information. In this context symmetric means that there exists a permuting unitary U such that $U^4 = \mathbb{1}$ and $\sigma_{BM}^{00} = U\sigma_{BM}^{01} = U^2\sigma_{BM}^{11} = U^3\sigma_{BM}^{10}$.

both of Alice's inputs with probability [52]

$$B_{OT}^{\text{pure}} \geq \frac{1}{4}\left(1+\frac{1}{2}\sqrt{1-2F}+\frac{1}{2}\sqrt{1+2F}\right)^2, \tag{9.12}$$

for $F \in [0, 1/2]$. Since there is no reason to bias Bob's ability to cheat based on Alice's choice of input, it seems likely that most protocols would output symmetric states and therefore, for protocols outputting pure states to Bob, the tighter bound would apply.

9.5.4 Security against Alice

Suppose Alice is dishonest and aims to guess the value of c output to Bob. In this section we present a cheating strategy that is always available to Alice, and which is always undetectable. We derive Alice's cheating probability given that she performs this strategy, and use this to obtain a lower bound for Alice's achievable cheating probability given that she performs some optimal strategy.

Let $|\Psi\rangle_{BMAE}$ be a purification of ρ_{BMA}, where E denotes the environment. Alice prepares an additional state $|+\rangle_D$ for use as a control qubit to perform her strategy. Since we consider information-theoretic security, Alice can do anything allowed by quantum mechanics and the overall state is

$$\frac{1}{\sqrt{2}}\left(|\Psi\rangle_{BMAE}|0\rangle_D + |\Psi\rangle_{BMAE}|1\rangle_D\right), \tag{9.13}$$

with Alice in complete control of systems A, E and D. Without loss of generality, we again assume that the two σ states with the highest fidelity are σ_{BM}^{00} and σ_{BM}^{01}. A valid cheating strategy available to Alice is as follows. In each Step 4 of the protocol, rather than performing a unitary $U_{MA}^{x_0x_1,i}$, Alice instead performs

$$U_{AM}^{00,i} \otimes |0\rangle\langle 0|_D + U_{AM}^{01,i} \otimes |1\rangle\langle 1|_D. \tag{9.14}$$

Defining the overall operations as $\mathcal{U} = V_{BM}^{(N)}U_{MA}^{00,N}\ldots V_{BM}^{(1)}U_{MA}^{00,1}$ and $\mathcal{V} = V_{BM}^{(N)}U_{MA}^{01,N}\ldots V_{BM}^{(1)}U_{MA}^{01,1}$, Alice's strategy leads to an output state

$$\begin{aligned}|\chi\rangle &:= \frac{1}{\sqrt{2}}\left(\mathcal{U}|\Psi\rangle_{BMAE}|0\rangle_D + \mathcal{V}|\Psi\rangle_{BMAE}|1\rangle_D\right)\\ &:= \frac{1}{\sqrt{2}}\left(|\psi^{00}\rangle_{BMAE}|0\rangle_D + |\psi^{01}\rangle_{BMAE}|1\rangle_D\right).\end{aligned} \tag{9.15}$$

This strategy is not detectable by Bob, since without access to system D it is as

if Alice has performed either the $x = 00$ or $x = 01$ honest operations, each with probability 1/2.

The states $\left|\psi^{jk}\right\rangle$ are purifications of σ_{BM}^{jk}, and all purifications are related by a unitary operation acting on the purifying system alone. As such, Alice is able to perform the further unitary operation

$$W_{AE}^{(1)} \otimes |0\rangle\langle 0|_D + W_{AE}^{(2)} \otimes |1\rangle\langle 1|_D , \tag{9.16}$$

where $W_{AE}^{(1)}$ and $W_{AE}^{(2)}$ are chosen to transform $|\psi^{00}\rangle$ and $|\psi^{01}\rangle$ into $|\phi^{00}\rangle$ and $|\phi^{01}\rangle$, such that the latter two states are the purifications of σ_{BM}^{00} and σ_{BM}^{01} with the highest overlap. This operation is performed so that we can later use Uhlmann's theorem to express Alice's cheating probability in terms of F, as we shall see. The resulting state is

$$|\Phi\rangle := \frac{1}{\sqrt{2}} \left(\left|\phi^{00}\right\rangle_{BMAE} |0\rangle_D + \left|\phi^{01}\right\rangle_{BMAE} |1\rangle_D \right). \tag{9.17}$$

In Step 8 of the protocol, Bob performs the POVM $\{\Pi_{BM}^z\}_z$ on $|\Phi\rangle$, where $z \in \{0*, 1*, *0, *1\}$. Our aim is to discover how well Alice can distinguish between the outcomes $c = 0$ and $c = 1$ using a measurement on her D system. The state of system D following Bob's POVM is[3]

$$\mu_D = \frac{1}{2} \sum_{i,j,z} \left\langle \phi^i \right| \Pi_{MB}^z \left|\phi^j\right\rangle |j\rangle\langle i|_D , \tag{9.18}$$

where $i, j \in \{0, 1\}$, $z \in \{0*, 1*, *0, *1\}$ and for ease of notation we have identified $\phi^0 := \phi^{00}$ and $\phi^1 := \phi^{01}$.

Eqs. (9.6) and (9.7) can be used to evaluate terms of the form $\langle\phi^i|\Pi_{BM}^z|\phi^i\rangle (= \langle\phi^{jk}|\Pi_{BM}^z|\phi^{jk}\rangle)$, since

$$\begin{aligned} \langle\phi^{jk}|\Pi_{BM}^z|\phi^{jk}\rangle &= \mathrm{Tr}_{BMAE}\Big(\Pi_{BM}^z \left|\phi^{jk}\right\rangle \left\langle\phi^{jk}\right| \Big) \\ &= \mathrm{Tr}_{BM}(\Pi_{BM}^z \sigma_{BM}^{jk}). \end{aligned} \tag{9.19}$$

The expression for μ_D can be further simplified using the following lemma.

Lemma 9.5. *For all values of $z \in \{0*, 1*, *0, *1\}$ and $jk \in \{00, 01, 11, 10\}$ such*

[3]Of course, since Bob's POVM acts on systems B and M only, the reduced state of the D system is unchanged regardless of whether Bob performs his measurement or not. Nevertheless, the D system is correlated with systems B and M, and this fact can be exploited by Alice to help her cheat, as we shall see.

that $\mathrm{Tr}_{BM}(\Pi^z_{BM}\sigma^{jk}_{BM}) = 0$*, it holds that*

$$(\Pi^z_{BM} \otimes \mathbb{1}_{AE}) \left|\phi^{jk}\right\rangle_{BMAE} = 0. \tag{9.20}$$

Proof. Since $\Pi^z_{BM} \otimes \mathbb{1}_{AE}$ is a positive semidefinite Hermitian operator, we can write its spectral decomposition as

$$\Pi^z_{BM} \otimes \mathbb{1}_{AE} = \sum_n c_n \left|c_n\right\rangle \left\langle c_n\right|, \tag{9.21}$$

where all c_n are positive real numbers. Therefore, using Eq. (9.19),

$$\begin{aligned} \mathrm{Tr}_{BM}(\Pi^z_{BM}\sigma^{jk}_{BM}) = 0 &\Rightarrow \langle\phi^{jk}|\Pi^z_{BM} \otimes \mathbb{1}_{AE}|\phi^{jk}\rangle = 0 \\ &\Rightarrow \langle c_i|\phi^{jk}\rangle = 0 \quad \forall i, \end{aligned} \tag{9.22}$$

and the result follows. □

Using this lemma, μ_D simplifies to

$$\begin{aligned} \mu_D &= \frac{1}{2}\left[\frac{1}{2}|0\rangle\langle 0|_D + \langle\phi^{01}|\Pi^{0*}_{MB}|\phi^{00}\rangle|0\rangle\langle 1|_D + \langle\phi^{00}|\Pi^{0*}_{MB}|\phi^{01}\rangle|1\rangle\langle 0|_D + \frac{1}{2}|1\rangle\langle 1|_D\right] \\ &\quad + \frac{1}{2}\left[\frac{1}{2}|0\rangle\langle 0|_D + \frac{1}{2}|1\rangle\langle 1|_D\right] \\ &= \frac{1}{2}\mu_D^{c=0} + \frac{1}{2}\mu_D^{c=1}, \end{aligned} \tag{9.23}$$

where the first square bracket corresponds to Bob obtaining an outcome $c = 0$ (i.e. Π^{0*} or Π^{1*}) and the second square bracket corresponds to Bob getting an outcome of $c = 1$ (i.e. Π^{*0} or Π^{*1}). Lastly, we must evaluate $\langle\phi^{01}|\Pi^{0*}_{MB}|\phi^{00}\rangle$.

To satisfy no-signalling, the density matrix in system D must be the same regardless of whether or not Bob actually performs his measurement. If Bob performs no measurement, Eq. (9.17) gives system D as

$$\frac{1}{2}[|0\rangle\langle 0|_D + \langle\phi^{01}|\phi^{00}\rangle|0\rangle\langle 1|_D + \langle\phi^{00}|\phi^{01}\rangle|1\rangle\langle 0|_D + |1\rangle\langle 1|_D]. \tag{9.24}$$

Comparing Eqs. (9.23) and (9.24), we must have $\langle\phi^{01}|\Pi^{0*}_{MB}|\phi^{00}\rangle = \langle\phi^{01}|\phi^{00}\rangle$. The trace distance between $\mu_D^{c=0}$ and $\mu_D^{c=1}$ is therefore $|\langle\phi^{01}|\phi^{00}\rangle|$, meaning that Alice can

use her D system to distinguish $c = 0$ from $c = 1$ with probability

$$\frac{1}{2}\left(1 + |\langle\phi^{01}|\phi^{00}\rangle|\right) = \frac{1}{2}\left(1 + F(\sigma^{00}_{BM}, \sigma^{01}_{BM})\right) := \frac{1}{2}\left(1 + F\right), \tag{9.25}$$

where the second equality follows from Uhlmann's theorem [170] since $|\phi^{00}\rangle$ and $|\phi^{01}\rangle$ are the purifications of σ^{00}_{BM} and σ^{01}_{BM} with maximum overlap.

9.5.5 Result

Previously, the best known lower bound for the cheating probabilities in 1-2 quantum OT was

$$\max\{A_{OT}, B_{OT}\} \geq 2/3. \tag{9.26}$$

Our results in the preceding section reproduce this bound, since

$$\begin{aligned} A_{OT} \geq \frac{1}{2}(1+F), \;\; B_{OT} \geq 1 - F \\ \Rightarrow \max\{A_{OT}, B_{OT}\} \geq \frac{2}{3}. \end{aligned} \tag{9.27}$$

Further, if the output states of the protocol are pure and symmetric, then we can use Eq. (9.12) to obtain the tighter bound

$$\max\{A_{OT}, B_{OT}\} \gtrapprox 0.749. \tag{9.28}$$

If instead we are particularly interested in one of either A_{OT} or B_{OT}, our construction quantifies the trade-offs possible between these parameters. This situation arises in the context of quantum signatures [110], where, in the distribution stage, signing keys are partially distributed in a manner very similar to 1-2 OT. In these protocols A_{OT} is prioritised, and it is important that $A_{OT} \approx 0.5$ to protect against repudiation attempts (see Section 3.7). On the other hand, to protect against forging attempts is much simpler, and the requirements on B_{OT} are less strict. The parametrisation of A_{OT} in terms of F suggests that in order to create an imperfect 1-2 OT scheme with a small ϵ_A, it is necessary to have a protocol which, in the honest case, outputs states that are almost orthogonal. Unfortunately, given $A_{OT} \approx 0.5$, our results show that it is necessary to have $B_{OT} \approx 1$. Therefore imperfect OT protocols will most likely not prove useful for quantum signatures in the information-theoretic security setting. Nevertheless, while imperfect OT has not proved useful for quantum signatures, there may be other useful direct applications.

9.6 Unambiguous Measurements

Classical-quantum states of the form $\rho_{XA} = \sum_{x \in \mathcal{X}} p(x) |x\rangle \langle x|_X \otimes \rho_A^x$ have been widely studied in quantum information in a variety of contexts such as channel coding, secure multiparty computations, quantum key distribution and quantum signatures to name a few. They can occur when quantum states (in this case ρ_A^x) are used to transmit classical information (in this case x). Retrieving the information stored in ρ_A^x using an "optimal" measurement is a subjective concept, and the identity of the optimal measurement depends heavily on the application. For communication protocols, it is common for the optimal measurement to be a minimum-error measurement – one which decodes the classical message with the smallest probability of error. For cryptographic protocols, the optimal measurement is often one which returns the largest possible amount of information while simultaneously disturbing the system less than a threshold amount.

A particular class of measurements we are interested in is unambiguous measurements. These measurements give "perfect" information in the sense that, given a successful measurement outcome, one can be certain that the decoded classical information is correct. Unambiguous measurements come in two main flavours: unambiguous state discrimination (USD), and unambiguous state elimination (USE). A successful USD measurement on ρ_A^x would identify x with certainty, but successful measurement outcomes do not occur with probability 1. USE measurements on the other hand can often be successful with probability 1, but only guarantee that $x \notin \mathcal{Y} \subset \mathcal{X}$, i.e. the measurement rules out states rather than definitively identifying the state. Intuitively, it seems that unambiguous measurements are well suited to cryptographic applications – their ability to provide "perfect yet partial" information on the states being sent is often exactly what is needed. More concretely, USD can be seen as very similar to Rabin OT, in which it is desired that the receiver obtains the sender's message with probability 1/2, and otherwise receives nothing with probability 1/2. On the other hand, USE measurements seem closely related to the more common 1-2 OT, in which incomplete but correct information is gained with certainty. Since OT plays a central role in secure two-party computations, it seems likely that unambiguous measurements could also play a major role in the developing field.

9.6.1 Semi-random OT using USE

In this section we present an interesting application of USE measurements. We describe a protocol for implementing many runs of Semi-random OT and analyse its security in the asymptotic limit. We again work in the information–theoretic security setting but this time prove *upper* bounds on the average cheating probabilities achievable for Alice and Bob in this protocol.

Note that the results in this section apply only to the cheating probabilities achievable for Alice and Bob *averaged* across all OT instances generated by the protocol. We make no claims regarding the cheating probabilities achievable on any single OT instance performed within the protocol. For this reason, the scheme in this section is not directly comparable to many existing OT schemes proposed in the literature which focus on performing a single instance of OT. Indeed, one must be very careful when trying to extend results on averaged cheating probabilities to worst-case bounds on any single OT instance, and we do not consider it here.

Nevertheless, if one considers the potential applications of imperfect OT, such as USS schemes, then having many instances of OT is exactly what is needed, and the important security parameter is the average cheating probability across all OT instances (see, for example, the Distribution Stage V2 in Section 3.7). It is conceivable that this would also be the case in other applications in which imperfect OT is used as a component within a larger protocol. We show that our protocol performs better than all previous protocols in terms of the average cheating probabilities it achieves across many OT instances. We further show that the average cheating probabilities are almost equal to the single-instance bounds derived in the previous section.

The protocol proceeds as follows:

1. Alice uniformly, randomly and independently selects N elements from the set $X = \{00, 01, 11, 10\}$. She encodes elements as $00 \rightarrow |00\rangle$, $01 \rightarrow |++\rangle$, $11 \rightarrow |11\rangle$ and $10 \rightarrow |--\rangle$.

2. Alice sends the N two-qubit states to Bob.

3. Bob randomly selects $\sqrt{N}$ out of the N states he receives and asks Alice to reveal their identity [4]. If Alice declares $|++\rangle$ or $|--\rangle$, then Bob measures both qubits in the X basis, otherwise he measures both qubits in the Z basis. The protocol aborts if any measurement result does not match Alice's declaration.

[4]The choice of $\sqrt{N}$ test bits is somewhat arbitrary. For security, we only need Bob to choose a number of test states such that: the number of test states tends to infinity as N increases; and the fraction of states chosen for testing tends to zero as N increases.

4. The $\sqrt{N}$ states selected in the previous step are discarded.

5. For each of the $N - \sqrt{N}$ remaining states, Bob measures the first qubit in the Z basis and the second qubit in the X basis. These measurements consitute two USE measurements (for example, an outcome of $|0\rangle$ on the first qubit rules out $|11\rangle$). Following these measurements, Bob can with certainty rule out one element from the set $Y_0 = \{00, 11\}$, and one from the set $Y_1 = \{01, 10\}$. In this way, for each of the remaining states he can know with certainty exactly one of x_0 and x_1, but not both (for example, if 11 and 10 are ruled out, then Bob knows that $x_0 = 0$).

The result of this protocol is that Alice and Bob have performed $N - \sqrt{N}$ runs of Semi-random OT, each of which could be used to implement a single instance of 1-2 OT, as per the construction in the Section 9.4. Below we analyse the average cheating probabilities achieved across all instances of Semi-random OT generated by this protocol. We show that this protocol can be made secure with average cheating probabilities (across all $N - \sqrt{N}$ instances) of $A_{OT} = 0.75$ and $B_{OT} \approx 0.729$.

9.6.2 Security against Bob

On each instance of OT, if Bob wants to cheat then he is successful if he correctly guesses both x_0 and x_1. In the asymptotic limit, the fraction of states discarded for testing in Step 3 tends to zero. Since the states are prepared independently, any strategy Bob performs (including general measurements correlated across all received states) cannot have an *average* success probability (probability of correctly identifying both x_0 and x_1) which is greater than the minimum-error measurement on a single unknown state taken from the set $S = \{|00\rangle, |++\rangle, |11\rangle, |--\rangle\}$. If there were such a measurement, Bob could simulate this strategy when he has only a single state from S and beat the known minimum-error measurement.

More concretely, suppose that when Alice sends Bob N states chosen independently from S, there exists a measurement $\mathcal{M}$ that Bob can make (potentially correlated across all N states) which leads to an average success probability (across all N states) that is greater than the success probability of the minimum-error measurement performed on a single state from S. Since the N states are chosen randomly and independently, if such a measurement existed then when Alice sends Bob only a single state from S, Bob could beat the single state minimum-error measurement by randomly creating a further $N - 1$ states himself and performing the measurement

$\mathcal{M}$ on the resulting N states. Of course, this is a contradiction by definition of the minimum-error measurement.

The above arguments do not show that correlated measurements provide no advantage. Correlated measurements performed across multiple states can be used to generate higher success probabilities on particular instances *if* one also allows for post-selection by Bob. Post-selection strategies are powerful and are the reason why the fraction of states used for testing must tend to zero. In our case though, since the fraction of test states tends to zero, Bob is effectively trying to optimally cheat on *almost all* received states. Therefore, in the asymptotic limit post-selection is irrelevant and we can bound Bob's average cheating probability across all $N - \sqrt{N}$ OT instances by considering the minimum-error measurement on a single state. Since the set S is a set of symmetric pure states, the minimum-error measurement is the SRM [52]. Using this measurement Bob can guess both of Alice's input bits with probability

$$B_{OT} = \frac{1}{4}\left(1 + \frac{1}{\sqrt{2}}\right)^2 \approx 0.729. \tag{9.29}$$

In this case, Bob's optimal strategy is the exact strategy considered in the general setting in Section 9.5.3.

9.6.3 Security against Alice

On each instance of OT, if Alice wants to cheat then her aim is to correctly guess the value of c such that Bob received x_c. To do this, she may send states other than the ones in S. In general, for the overall protocol Alice will generate $\rho_{AB_{11}B_{12}B_{21}B_{22}\ldots B_{N1}B_{N2}}$ and send the B systems to Bob, keeping the A system for herself. In Step 3 of the protocol Bob then randomly selects pairs of the states he received, say $\rho_{B_{k1}B_{k2}}$, and asks Alice to declare the identity of the state. He does this for $\sqrt{N}$ of the N pairs. Since we are looking for an upper bound on Alice's capabilities, we assume that she holds a purification $|\Psi\rangle_{B_{k1}B_{k2}A}$ of $\rho_{B_{k1}B_{k2}}$.

Alice must declare a state to Bob that will agree with his measurement outcomes in Step 3. If she can do this with certainty, then the state $|\Psi\rangle_{B_{k1}B_{k2}A}$ must be of the form

$$\begin{aligned} |\Psi\rangle_{B_{k1}B_{k2}A} = b_0|00\rangle_{B_{k1}B_{k2}}|0\rangle_A + b_1|++\rangle_{B_{k1}B_{k2}}|1\rangle_A \\ + b_2|11\rangle_{B_{k1}B_{k2}}|2\rangle_A + b_3|--\rangle_{B_{k1}B_{k2}}|3\rangle_A, \end{aligned} \tag{9.30}$$

where $\{|0\rangle_A, |1\rangle_A, |2\rangle_A, |3\rangle_A\}$ is an orthonormal set. If Alice does not send states in the above form, then she cannot guess Bob's measurement outcomes with certainty,

and for asymptotically large N it becomes virtually certain that the protocol will abort.

Essentially, this means that Alice is restricted to the attacks considered in the general protocol analysis in Section 9.5.4 – attacks in which she sends superpositions of honest states. In fact, it is numerically verifiable that an optimal strategy for Alice is to prepare

$$\frac{1}{\sqrt{2}}\left(|00\rangle_B|0\rangle_A + |++\rangle_B|1\rangle_A\right), \tag{9.31}$$

which corresponds exactly to the operation given in Eq. (9.14). Since the overlap between all adjacent states in S is $1/2$, Eq. (9.25) implies that Alice can correctly guess the value of c with probability 0.75. In fact, this argument shows that Alice's probability of guessing c is at most 0.75 for *all* non-test instances of OT within the protocol.

9.7 Conclusion

In this chapter we introduced Semi-random OT and a general framework useful for its study. We explicitly constructed undetectable cheating strategies available to Alice and Bob and used them to lower bound the cheating probability p_C of any Semi-random OT protocol. Section 9.4 implies that the derived bounds are directly transferable to standard 1-2 quantum OT, allowing us to reproduce the known lower bound $p_C \geq 2/3$, or, if the states output by the honest protocol are pure and symmetric, improve the bound to $p_C \geq 0.749$.

In applications more sensitive to sender dishonesty than receiver dishonesty (or vice versa), our parametrisation of A_{OT} and B_{OT} in terms of the fidelity shows explicitly how reductions in one party's ability to cheat will impact the other's cheating probability. This relationship proves useful in the context of quantum signatures, where it is desirable to have $A_{OT} \approx 0.5$ but the requirements on B_{OT} are less strict.

Chapter 10

Secret-key quantum money

10.1 Introduction

Quantum money was the first example of a cryptographic protocol using quantum mechanics to provide distinct advantages over all classical protocols. Originally suggested by Weisner in 1970 [154], the basic aim of any quantum money scheme is to enable a trusted authority, the bank, to provide untrusted users with finitely re-usable, verifiable coins that cannot be forged. Verifiability ensures that honest users can prove that the money they hold is genuine, while unforgeability restricts the ability of an adversary to dishonestly fabricate additional coins.

These schemes are tangentially related to signature schemes, insofar as there are potentially many participants sending/receiving tokens which must be unforgeable and finitely transferable. However, there are also significant differences. Most notably, the bank is a trusted participant of a quantum money scheme, whereas the sender is untrusted in USS schemes. As such, for quantum money schemes, both their construction and the types of dishonest behaviour available to an adversary are markedly different to signatures. Nevertheless, many of the techniques used in the security analysis of USS schemes are transferable.

10.2 Related work

Weisner's original quantum money scheme contained two major drawbacks, namely: verification required quantum communication between the holder and the bank; and the security of the scheme was not rigorously defined or proved. Indeed, it was shown in Refs. [171–173] that many variants of the scheme were vulnerable to so-called "adaptive attacks" – attacks in which the adversary is allowed a number of auxiliary

interactions with the bank before trying to forge a coin.

In 2012, Gavinsky [174] addressed both issues and presented a fully secure quantum money scheme in which coins are verified using three rounds of *classical* communication between the holder of the coin and the bank. The scheme was based on hidden matching quantum retrieval games (QRGs), first introduced in Ref. [175]. Nevertheless, the scheme could not be considered *practical*, as the security analysis did not include the effects of noise. This issue was addressed by Pastawski et al. [176], in which a noise tolerant quantum money scheme with classical verification was proposed that remains secure as long as the overall transmission fidelity is greater than $\frac{1}{2} + \frac{1}{\sqrt{8}} \approx 85.4\%$. The scheme requires only two rounds of communication for verification and is secure even against adaptive attacks. Following this, Ref. [177] presented a simpler protocol, again based on hidden matching QRGs, in which the verification procedure contained only a single round of communication and displayed an increased noise tolerance of up to 12.5%, where noise is defined as the probability of a single honest verifier measurement returning an incorrect outcome.

Beyond the secret-key quantum money schemes discussed above, there has also been significant interest in public-key quantum money schemes, first proposed in [171], offering computational security against quantum adversaries. Since then, Farhi et al. [178] introduced the concepts of quantum state restoration and single-copy tomography to further rule out a large class of seemingly promising schemes. Following this result, Farhi et al. [179] suggested a scheme based on knot theory and conjectured that it is secure against computationally bounded adversaries. However, whether a secure public-key quantum money scheme exists without the use of oracles is an open question and, so far, the majority of schemes that were proposed have subsequently been broken [180].

Our contributions

In this chapter, as always, we work in the information-theoretic security setting and focus on secret-key quantum money schemes with classical verification. We present a family of schemes, based on hidden matching quantum retrieval games (QRGs), which display a number of benefits over previous proposals. First, our schemes are more noise/error tolerant than all previous proposals; our schemes can tolerate noise up to 23%, which we conjecture reaches 25% asymptotically as the dimension of the underlying hidden matching states is increased. Furthermore, we prove that 25% is the maximum tolerable noise for a wide class of quantum money schemes with classical verification, meaning our schemes are almost optimally noise tolerant.

We use methods in semi-definite programming to prove security in a substantially different manner to previous proposals [174, 177], leading to two main advantages: first, coin verification involves only a constant number of states (with respect to coin size), thereby allowing for smaller coins; second, the re-usability of coins within our scheme grows linearly with the size of the coin, which is known to be optimal. Finally, we discuss how our schemes can be implemented in practice using a coherent state encoding, while also showing that they remain secure even in the presence of limited detection efficiency. The work presented in this chapter has been published in Ref. [181] with minor modifications.

10.3 Definitions

In this section we state various definitions that are needed to introduce our quantum money schemes. We consider the case of quantum money "mini-schemes" in which the bank creates only a single quantum coin and the adversary attempts to use this coin to forge another copy. It has been shown in Ref. [182] that by adding a classical serial number to each coin, a secure full quantum money scheme can be created directly from the secure mini-scheme, and so the two are essentially equivalent.

Definition 10.1. A quantum money mini-scheme with classical verification consists of an algorithm, Bank, which creates a quantum coin \$ and a verification protocol Ver, which is a classical protocol run between a holder H of \$ and the bank B, designed to verify the authenticity of the coin. The final output of this protocol is a bit $b \in \{0, 1\}$ sent by the bank, which corresponds to whether the coin is valid or not. Denote by $\mathsf{Ver}_H^B(\$)$ this final bit. The scheme must satisfy two properties to be secure:

- Correctness: The scheme is ϵ-correct if for every honest holder, we have

$$\mathbb{P}[\mathsf{Ver}_H^B(\$) = 1] \geq 1 - \epsilon.$$

- Unforgeability: Coins in the scheme are ϵ-unforgeable if for any quantum adversary who has interacted a finite and bounded number of times with the bank and holds a valid coin \$, the probability that she can produce two coins $\$_1$ and $\$_2$ that are verified by an honest user satisfies

$$\mathbb{P}\left[\mathsf{Ver}_H^B(\$_1) = 1 \wedge \mathsf{Ver}_H^B(\$_2) = 1\right] \leq \epsilon,$$

where H is any honest holder.

The first property guarantees that all honest participants can prove the coins they own are valid, while the second property guarantees that a dishonest adversary cannot forge the coins. The definition covers adaptive attacks by allowing the adversary to interact with the bank (via the verification procedure) a finite number of times before attempting to forge the coin.

The schemes presented in this chapter are based on quantum retrieval games (QRGs), which we have mentioned but not formally introduced. A QRG is a protocol performed between two parties, Alice and Bob, and can be seen as a generalisation of state discrimination. Alice holds an n-bit string x, selected at random according to a probability distribution $p(x)$, which she encodes into a quantum state ρ_x. She sends the state to Bob, whose goal is to provide a correct answer to a given question about x. Mathematically, a question is modelled as a relation: if X is the set of possible values x can take, and if A is the set of possible answers, the relation σ is a subset of $X \times A$. If $(x, a) \in \sigma$, this means that, given x, the answer a is a correct answer to the "question" σ. Formally, a quantum retrieval game is defined as follows.

Definition 10.2. Let X and A be the sets of inputs and answers respectively. Let $\sigma \subset X \times A$ be a relation and $\{p(x), \rho_x\}$ an ensemble of states and their a priori probabilities. Then the tuple $G = (X, A, \{p(x), \rho_x\}, \sigma)$ is called a quantum retrieval game. If Bob may choose to find an answer to one of a finite number of distinct relations $\sigma_1, ..., \sigma_k$, then we write the game as $G = (X, A, \{p(x), \rho_x\}, \sigma_1, ..., \sigma_k)$.

A particularly useful class of QRGs are the *hidden matching* QRGs [174, 177, 183], in which the relations are defined by matchings. A matching M on the set $[n] := \{1, 2, ..., n\}$, where n is an even number, is a partitioning of the set into $n/2$ disjoint pairs of numbers[1]. A matching can be visualised as a graph with n nodes, where edges define the elements in the matching, as illustrated in Fig. 10.1. In general, there are $1 \times 3 \times \ldots \times (n-1) = (n-1)!!$ distinct matchings of any set containing n elements. For our purposes, we focus on sets of matchings where no two matchings in the set contain a common element. We call such sets *pairwise disjoint*. The maximum number of pairwise disjoint matchings is $n - 1$, since if we consider the element $1 \in [n]$, it must be paired in each matching with a distinct integer less than or equal to n.

[1]More precisely, this is actually the definition of a *perfect* matching.

Definition 10.3. A maximal pairwise disjoint set of matchings, $\mathcal{R}$, is a set of pairwise disjoint matchings on $[n]$ such that $|\mathcal{R}| := n - 1$.

A matching on the set $[n]$ can be equivalently represented as a graph with n nodes, with each each element (i, j) of the matching identified with an edge in the graph. Maximal pairwise disjoint sets of matchings for $n = 4, 6$, and 8 are illustrated in Fig. 10.1.

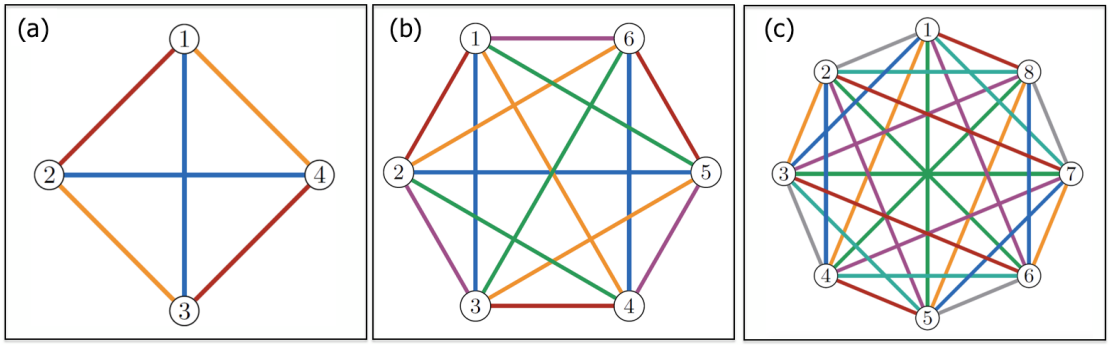


Figure 10.1: Maximal pairwise disjoint set of matchings for (a) $n = 4$, (b) $n = 6$ and (c) $n = 8$. Colour is used to represent each matching within the maximal pairwise disjoint set.

In hidden matching QRGs the set of possible inputs is the set of all n-bit strings, each chosen with equal probability, where n is an even number. Alice encodes her input into the n-dimensional pure state

$$|\phi_x\rangle = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} (-1)^{x_i} |i\rangle \tag{10.1}$$

where x_i is the i-th bit of the string x. Note that this state corresponds to a $O(\log_2 n)$ qubit state, so that the number of qubits needed in the scheme scales favourably with n.

The relations in this game are defined by the matchings: given a matching, the correct answers are the ones which correctly identify the parity of the bits connected by an edge in the matching. For example, if $(1, 2)$ is an element of the matching, the measurement should output $x_1 \oplus x_2$. Formally, given a perfect matching M_1, the set of answers is given by

$$A = \{(i, j, b) : i, j \in \{1, ..., n\}, b \in \{0, 1\}\}$$

and the corresponding relation is

$$\sigma_1 = \{(x, i, j, b) : x_i \oplus x_j = b \text{ and } (i, j) \in M_1\}.$$

Bob is able to find a correct answer to any matching of his choice with certainty simply by measuring in the basis

$$\mathcal{B} = \{\frac{1}{\sqrt{2}}(|i\rangle \pm |j\rangle)\}, \quad \text{with } (i,j) \in M. \tag{10.2}$$

This is because the outcome $\frac{1}{\sqrt{2}}(|i\rangle + |j\rangle)$ can only occur if $x_i \oplus x_j = 0$, and similarly $\frac{1}{\sqrt{2}}(|i\rangle - |j\rangle)$ can only occur if $x_i \oplus x_j = 1$.

Previous quantum money schemes based on hidden matching QRGs have used only two matchings for verification. In the following section, we generalise these schemes to the case of an arbitrary number of matchings and show that this allows us to significantly increase the noise tolerance of the resulting schemes.

10.4 Quantum money scheme

Here we present a quantum money scheme which is secure even in the presence of up to 23% noise. As in Ref. [177], the verification protocol requires only one round of classical communication.

In this scheme, the bank randomly chooses a number of n-bit classical strings and encodes each of them into the hidden matching states, given by Eq. (10.1). Essentially, the coin is a collection of these independent quantum states, and each of the quantum states can be thought of as an instance of a QRG. We assume that there is a maximal pairwise disjoint set of matchings on $[n]$, known to all participants, which we call $\mathcal{R}$. This set specifies the $n-1$ possible relations defined within each QRG, and each state in the coin represents a QRG. To verify a coin, the holder will pick a small selection of the states from the coin and randomly choose a relation for each. The holder will perform the appropriate measurement (defined by Eq. (10.2)) to get an answer for each QRG under each chosen relation. The holder then sends these answers to the bank which returns whether more than a specified fraction of the answers are correct or not. If they are, the coin is accepted as valid; otherwise, it is rejected. The scheme is formally defined below and illustrated in Figs. 10.2 and 10.3.

Bank Algorithm

1. The bank independently and randomly chooses q n-bit strings which we will call $x^1, ..., x^q$.

2. For $i \in [q]$, the bank creates $\phi_{x^i} := |\phi_{x^i}\rangle\langle\phi_{x^i}|$, where

$$|\phi_{x^i}\rangle := \frac{1}{\sqrt{n}} \sum_{j=1}^{n} (-1)^{x^i_j} |j\rangle.$$

For each i we define the QRG $G_i = (S_i, A_i, \{\phi_{x^i}\}_{x^i}, \sigma_1, ..., \sigma_{n-1})$, where $\mathcal{R} = \{\sigma_1, ..., \sigma_{n-1}\}$ is a maximal pairwise disjoint set of matchings known to all participants in the scheme.

3. The bank creates the classical binary register, r, and initialises it to 0^q.

4. The bank creates the counter variable s and initialises it to 0.

5. The pair $(\$, r) = (\bigotimes_{i=1}^{q} \phi_{x^i}, r)$ is the coin for the mini-scheme. The bank keeps the counter s in order to keep track of the number of verification attempts.

Ver Algorithm

1. The holder of the coin randomly chooses a subset of indices, $L \subset [q]$ such that $r_i = 0$ for each $i \in L$. The indices $i \in L$ specify the selection of games G_i which will be used as tests in the verification procedure. For each $i \in L$, the holder sets the corresponding bit of r to be 1 so that this game cannot be used in future verifications.

2. For each $i \in L$, the holder picks a relation σ'_i at random from $\mathcal{R}$ and applies the appropriate measurement to obtain outcome d_i.

3. The holder sends all triplets (i, σ'_i, d_i) to the bank.

4. The bank checks that $s < T$, where T is the pre-defined maximum number of allowed verifications for the coin. If $s = T$, the bank declares the coin as invalid.

5. For each i, the bank checks whether the answer is correct by comparing (i, σ'_i, d_i) to the secret x^i values. The bank accepts the coin as valid if and only if more than $l(c - \delta)$ of the answers are correct, where c is a correctness parameter of the protocol, $l = |L|$, and δ is a small positive constant.

6. The bank updates s to $s + 1$.

We say that an instance of the verification algorithm has been passed/failed if the final output by the bank is "valid"/"invalid" respectively. Coins can be verified at most T times until the Hamming weight of r is greater than Tl, at which point the

coin is returned to the bank to be refreshed. We choose T to be small but linear in q. Any such choice would be acceptable but, for the sake of definiteness, in what follows we set $T := q/(1000l)$. We note that having T scale linearly with q is optimal for any quantum money scheme [174] and that this is an improvement over previous protocols (for example those in Refs. [174, 177]).

The noise of the protocol is defined as the probability that an honest verifier obtains an incorrect outcome when making the honest measurement on a single QRG state (i.e. in step 2 of the verification procedure). In the ideal setting we can set $c = 1$, since an honest participant in possession of a correct state will always get a correct answer to a relation. Of course, in practice system imperfections inevitably lead to errors so that even when all participants are honest, it is not certain that the holder's measurement will return a correct answer. Thus, in the presence of errors, we must have $c < 1$, and the smallest value of c for which we can retain security determines the noise tolerance of the protocol.

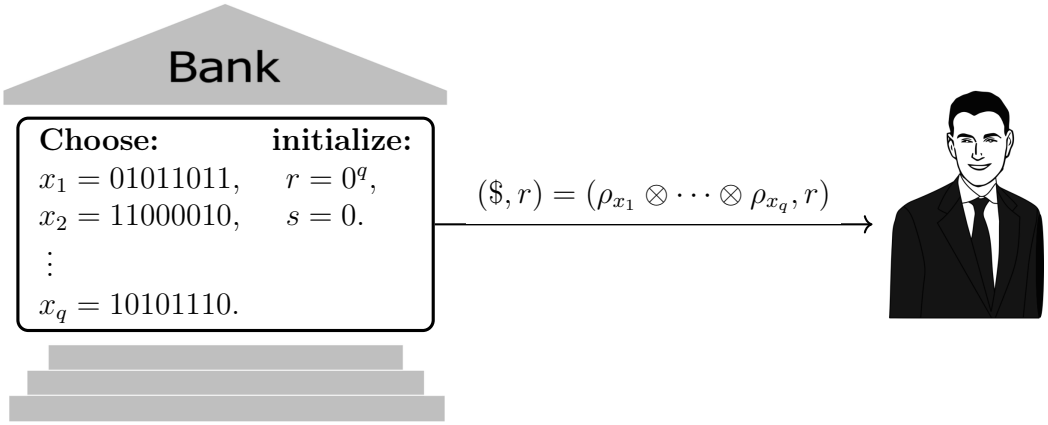


Figure 10.2: Schematic illustration of the Bank algorithm for $n = 8$. The bank selects q 8-bit strings and initializes the q-bit register r to the zero string. The bank creates the corresponding hidden matching states and sends these, together with r, to the holder of the coin.

We note that this scheme requires the bank to maintain a small classical database to record the number of times the verification protocol has been run – i.e. the bank's database is "non-static", and must be updated after each run of verification. Although this requirement demands more from the bank than completely static database models, we believe the requirement is both minimal and realistic, and allows significant simplifications to the security analysis.

Nevertheless, in some cases it may be desirable for the bank to have a completely static database – for example in applications in which the bank consists of many small decentralised branches. In such a scenario, attacks targeting multiple branch locations may be able to compromise security by gaining additional verification

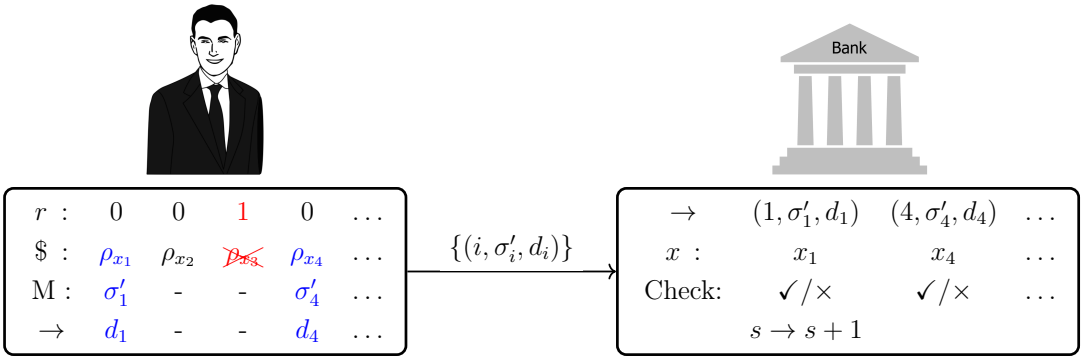


Figure 10.3: Schematic showing the verification algorithm. The verifier selects a sample $\{\rho_{x_1}, \rho_{x_4}, \ldots\}$ of the states contained within the coin which have an r value of 0. He randomly chooses matching measurements and applies them to get classical measurement outcomes which he sends to the bank, together with the index of the state and the matching chosen. The bank checks these against its secret strings, as well as checking $s < T$. Finally, the bank declares an output based on the number of incorrect outcomes.

attempts. To provide safeguards against these types of attack, our scheme could be modified in two different ways.

The simplest method would be to assume that all bank branches have access to a single common database, thereby preventing verifiers from performing too many verification attempts on a single coin. Alternatively, we could add an additional round of classical communication to the verification protocol, similarly to Ref. [174], in which the bank selects the states to be used in the verification protocol. The effect would be to transform our scheme into one which uses a fully static database, but still retains the same level of noise tolerance. Security of this modified scheme can be proved by directly applying the arguments in Ref. [174] to show that the additional verification attempts do not (significantly) help the adversary[2].

10.4.1 Security

In this section we prove that the scheme defined above is secure according to Definition 10.1.

Correctness

Correctness of the scheme follows simply from the Hoeffding bound [58]. In the honest case, if the holder of a coin has probability c of getting a correct answer for each of the l QRGs selected in the verification protocol, then his probability of

[2]We are able to apply the arguments in Ref. [174] because, although our scheme uses more than two matchings, when taken pairwise any two matchings within our scheme are independent.

getting fewer than $(c-\delta)l$ correct answers overall is bounded by

$$\mathbb{P}(\text{Honest Fail}) \leq e^{-2l\delta^2}. \tag{10.3}$$

Based on the security analysis in the following section, we choose δ to be half of the gap between the error rate an honest participant expects and the minimum error rate the adversary can achieve. I.e. we set $\delta := (e_{\min} - \beta)/2$, where $e_{\min}$ is the minimum error rate achievable by the adversary (derived below in Eq. (10.27)), and $\beta := 1 - c$ is the error rate expected in an honest run of the protocol.

Unforgeability

We assume the adversary is in possession of a valid coin and first address a simple forging strategy available to the adversary based on manipulating the r register attached to the coin. The adversary is allowed to set at most $q/1000$ of the r register entries to 1. She creates $(\$_1, r_1)$ and $(\$_2, r_2)$ to send to the two honest verifiers, Ver_1 and Ver_2 respectively. If she sets $r_1(i) = 1$ and $r_2(i) = 0$, she can be certain that Ver_1 will not select the i'th state to test, and so can forward the perfect state to Ver_2. In this way, $q/1000$ of the states in the coins sent to each verifier will be perfect, and will not cause errors. The remaining positions must have r register values of 0 for both verifiers. Similarly, the adversary is able to use the auxiliary verification attempts to her advantage. We make a worst-case assumption and assume that the adversary gets full knowledge of every state used in an auxiliary verification attempt. Since there are at most T attempts allowed, each of which involve l states, the adversary knows the identity of at most $q/1000$ of the states. Since the states are prepared independently, this knowledge does not provide any information on the remaining states.

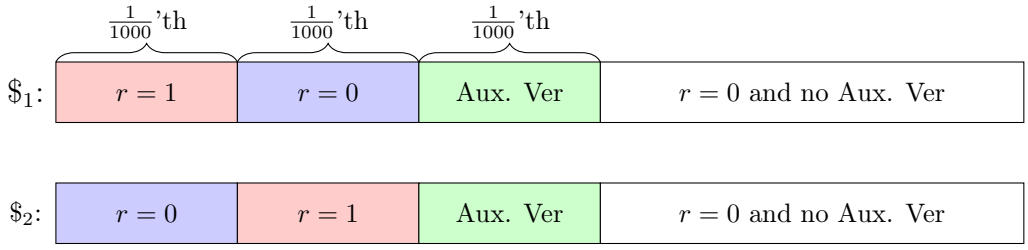


Figure 10.4: Representation of the states within the quantum coins sent to the verifiers. The first block on the far left represents all states for which the adversary set $r = 1$ for Ver_1, and $r = 0$ for Ver_2. The adversary knows that Ver_1 cannot select these states for testing, and so is able to forward on the perfect states to Ver_2. The second block of states represents the same, but with the roles of the verifiers reversed. The Aux. Ver states in the diagram are the ones that we assume are known to the adversary via auxiliary verifications. The remaining states in white are the ones we consider below – those states for which the r register is zero for both verifiers, and which have not been used in auxiliary verifications.

The combined effect of the above two strategies is that the adversary is able to exactly replicate $q/500$ of the states in the coin, as shown in Fig. 10.4. To prove coins are unforgeable, we consider the remaining $997q/1000$ states for which the r register is zero for both verifiers, and for which the adversary has no auxiliary information. In reference to Fig. 10.4, we refer to these states as the white states, and start by considering a single such state, $\phi_{x^i} := |\phi_{x^i}\rangle \langle \phi_{x^i}|$, contained in the coin. For simplicity, we drop the superscript on the n-bit strings x^i in all that follows.

The idea behind the proof is to relate the probability that the forger can use a single white state to create two states that pass the verification test of the two honest verifiers, to the average fidelity of these two states with the original state $|\phi_x\rangle$. The maximisation of this average fidelity corresponds to the optimal attack, which can be cast as a semi-definite program. By focusing on the dual program, we can upper bound the value of the semi-definite program and therefore bound the forging probability of the adversary. Lastly, we show that coherent attacks on multiple states cannot help the adversary to forge.

Since the adversary has a valid coin, she holds the unknown state

$$|\phi_x\rangle = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} (-1)^{x_i} |i\rangle. \tag{10.4}$$

From this state, the adversary wishes to create two states, η_x and τ_x, which, when measured by the honest verifiers, will give the correct answer to a randomly chosen relation in $\mathcal{R}$. At this stage we ignore any auxiliary verification attempts available to her. Consider the normalised state sent to Ver_1,

$$\eta_x = \sum_{i,j=1}^{n} a_{ij} |i\rangle\langle j|. \tag{10.5}$$

Suppose the verifier chooses to measure using the matching $M_\alpha = \{(i_1, j_1), ..., (i_{n/2}, j_{n/2})\}$, where $\alpha \in \{1, 2, \ldots, n-1\}$. To find a correct answer to the relation σ_α defined by this matching, an honest verifier will apply the measurement with projectors in the set $\{|+_{i_k j_k}\rangle \langle +_{i_k j_k}|, |-_{i_k j_k}\rangle \langle -_{i_k j_k}| \ : \ k = 1, ..., n/2\}$, where $|\pm_{i_k j_k}\rangle := \frac{1}{\sqrt{2}}(|i_k\rangle \pm |j_k\rangle)$. An incorrect result is obtained whenever the verifier finds an incorrect value for $x_{i_k} \oplus x_{j_k}$, which happens whenever the measurement outcome is one of the form

$$\frac{1}{\sqrt{2}}(|i\rangle - (-1)^{x_i \oplus x_j} |j\rangle). \tag{10.6}$$

This happens with probability

$$p^{\alpha,x}_{\mathrm{Ver}_1} = \frac{1}{2}\left(1 - \sum_{k=1}^{n/2}(-1)^{x_{i_k}\oplus x_{j_k}} a_{i_k j_k} + (-1)^{x_{i_k}\oplus x_{j_k}} a_{j_k i_k}\right). \tag{10.7}$$

Thus, the probability of an incorrect answer to σ_α is given by a subset of the off-diagonal elements of the density matrix η_x. The off-diagonal elements occurring are exactly those with indices paired by the matching M_α. Since the set of relations form a maximal pairwise disjoint set, the off-diagonal matrix elements appearing in the error probability for different relations will all be distinct. Therefore, averaging over all possible relations that could be chosen by the verifier allows us to significantly simplify the adversary's error probability, which becomes

$$\begin{aligned} p^{x}_{\mathrm{Ver}_1} &= \frac{1}{n-1}\sum_{\alpha=1}^{n-1} p^{\alpha,x}_{\mathrm{Ver}_1} = \frac{1}{2(n-1)}\left(n - \sum_{i,j=1}^{n}(-1)^{x_i\oplus x_j} a_{ij}\right) \\ &= \frac{n}{2(n-1)}(1 - F_x), \end{aligned} \tag{10.8}$$

where we have defined

$$F_x := \langle\phi_x|\eta_x|\phi_x\rangle = \frac{1}{n}\sum_{i,j}(-1)^{x_i\oplus x_j} a_{ij}. \tag{10.9}$$

Since the adversary does not know the secret string x, rather than holding the state in Eq. (10.4), she instead holds a mixture over the possible x values. We define $F := \frac{1}{2^n}\sum_x F_x$ and take an average over x values to get

$$p_{\mathrm{Ver}_1} = \frac{1}{2^n}\sum_x p^x_{\mathrm{Ver}_1} = \frac{1}{2^n}\sum_x \frac{n}{2(n-1)}(1 - F_x) = \frac{n}{2(n-1)}(1 - F). \tag{10.10}$$

Essentially then, to successfully forge a coin, the adversary is trying to create two states, η_x and τ_x, which both have a high fidelity with the original state $|\phi_x\rangle$. Let's define $G_x = \langle\phi_x|\tau_x|\phi_x\rangle$, and $G := \frac{1}{2^n}\sum_x G_x$. For the purpose of forging, the adversary needs *both* Ver_1 and Ver_2 to accept the coin she sends, which requires her to make both error probabilities as small as possible. From the above result, we can relate this to maximising the average fidelity of the states η_x and τ_x with the original state. This problem can be cast as a semi-definite program as follows.

Let $\Psi : L(\mathcal{X}) \to L(\mathcal{Y}\otimes\mathcal{Z})$ be a physical channel taking states in Hilbert space $\mathcal{X}$ to states in the Hilbert space $\mathcal{Y}\otimes\mathcal{Z}$, where both $\mathcal{Y}$ and $\mathcal{Z}$ are isomorphic to $\mathcal{X}$.

We want to find the channel that maximises

$$\overline{F} = \frac{1}{2^n} \sum_{x=1}^{2^n} \frac{\langle \phi_x | \eta_x | \phi_x \rangle + \langle \phi_x | \tau_x | \phi_x \rangle}{2}, \tag{10.11}$$

where $\eta_x = \mathrm{Tr}_{\mathcal{Z}} \left[\Psi(|\phi_x\rangle\langle\phi_x|) \right]$ and $\tau_x = \mathrm{Tr}_{\mathcal{Y}} \left[\Psi(|\phi_x\rangle\langle\phi_x|) \right]$. In other words, η_x is the reduced state of the channel output representing the state held by Ver_1, and τ_x is the reduced state of the channel output representing the state held by Ver_2. This maximisation is subject to Ψ being a completely positive trace preserving linear map. To express this maximisation in the standard form of a semi-definite program, we express the channel as an operator using the Choi representation. We fix the preferred basis to be $\{|i\rangle\}_{i=1,\ldots,n}$, the basis used to define the hidden matching states in the ensemble. Given this choice, the Choi operator corresponding to the channel Ψ is an operator $J(\Psi)$ in $L(\mathcal{X} \otimes \mathcal{Y} \otimes \mathcal{Z})$, given by

$$J(\Psi) = \sum_{i,j=1}^{n} |i\rangle\langle j|_{\mathcal{X}} \otimes \Psi(|i\rangle\langle j|)_{\mathcal{YZ}} \tag{10.12}$$

Using the facts that $\langle \phi_x | i \rangle = \langle i | \phi_x \rangle$ for all states in the ensemble, and that Ψ is a linear map, it can be shown that

$$\mathrm{Tr}_{\mathcal{XYZ}} \left[\left(\phi_x^{\mathcal{X}} \otimes \phi_x^{\mathcal{Y}} \otimes \mathbb{1}^{\mathcal{Z}} \right) J(\Psi) \right] = \langle \phi_x | \eta_x | \phi_x \rangle_{\mathcal{Y}}, \tag{10.13}$$

and similarly that

$$\mathrm{Tr}_{\mathcal{XYZ}} \left[\left(\phi_x^{\mathcal{X}} \otimes \mathbb{1}^{\mathcal{Y}} \otimes \phi_x^{\mathcal{Z}} \right) J(\Psi) \right] = \langle \phi^x | \tau_x | \phi^x \rangle_{\mathcal{Z}}, \tag{10.14}$$

where here, for ease of notation, we have used the superscript to denote the relevant Hilbert space. With this we can rewrite the problem in Eq. (10.11) as the problem of finding the operator $J(\Psi)$ which maximises

$$\frac{1}{2^{n+1}} \sum_{x=1}^{2^n} \mathrm{Tr}_{\mathcal{XYZ}} \left[\left((\phi_x^{\mathcal{X}} \otimes \phi_x^{\mathcal{Y}} \otimes \mathbb{1}^{\mathcal{Z}}) + (\phi_x^{\mathcal{X}} \otimes \mathbb{1}^{\mathcal{Y}} \otimes \phi_x^{\mathcal{Z}}) \right) J(\Psi) \right]. \tag{10.15}$$

The conditions that the channel must be completely positive and trace preserving lead to the conditions that $J(\Psi)$ must be positive semidefinite and $\mathrm{Tr}_{\mathcal{YZ}}(J(\Psi)) = \mathbb{1}_{\mathcal{X}}$. Written in standard form, the semidefinite program corresponding to the max-

imum average fidelity is given by

$$\begin{aligned} &\text{Maximise:} && \langle Q(n), X\rangle \\ &\text{subject to:} && \mathrm{Tr}_{\mathcal{YZ}}(X) = \mathbb{1}_{\mathcal{X}} \\ & && X \geq 0, \end{aligned} \tag{10.16}$$

where

$$Q(n) = \frac{1}{2^{n+1}} \sum_{x=1}^{2^n} \Big((\phi_x^{\mathcal{X}} \otimes \phi_x^{\mathcal{Y}} \otimes \mathbb{1}^{\mathcal{Z}}) + (\phi_x^{\mathcal{X}} \otimes \mathbb{1}^{\mathcal{Y}} \otimes \phi_x^{\mathcal{Z}}) \Big). \tag{10.17}$$

The dual problem is simply

$$\begin{aligned} &\text{Minimise:} && \mathrm{Tr}(Y) \\ &\text{subject to:} && \mathbb{1}_{\mathcal{YZ}} \otimes Y \geq Q(n) \\ & && Y \in \mathrm{Herm}(\mathcal{X}), \end{aligned} \tag{10.18}$$

since $\langle \mathbb{1}_{\mathcal{X}}, Y\rangle = \mathrm{Tr}(Y)$ and the adjoint of the partial trace is the extension by the identity. The dual problem approaches the optimal value from above, so any feasible point (i.e. any operator Y that satisfies the constraints of the dual problem) gives us an upper bound on the maximum average fidelity. A feasible point can easily be found in terms of the matrix $Q(n)$ as

$$Y = ||Q(n)||_\infty \mathbb{1}_{\mathcal{X}} \tag{10.19}$$

so that we arrive at the following upper bound on the average fidelity:

$$\overline{F} \leq n||Q(n)||_\infty. \tag{10.20}$$

Thus, for quantum money protocols using states of dimension n and a maximal disjoint set of matchings, we can upper bound the error probability of the adversary in terms of the operator norm of $Q(n)$. Computing this norm for different values of n leads to the bound

$$\overline{F} \leq \frac{1}{2} + \frac{1}{n} \tag{10.21}$$

which we have verified numerically for $n \leq 14$ and we conjecture holds for any n. From now on, we simply assume that $n \leq 14$. The analysis above enables us to

restrict the achievable error probabilities for the two verifiers on a single game as

$$\begin{aligned} p_{\mathrm{Ver}_1} &= \frac{n}{2(n-1)}\,(1-F) \\ p_{\mathrm{Ver}_2} &= \frac{n}{2(n-1)}\,(1-G) \end{aligned} \tag{10.22}$$

subject to

$$\frac{1}{2}(F+G) \leq \frac{1}{2} + \frac{1}{n}, \tag{10.23}$$

which leads to

$$p_{\mathrm{Ver}_1} + p_{\mathrm{Ver}_2} \geq \frac{1}{2} - \frac{1}{2(n-1)}. \tag{10.24}$$

Until now, we have considered only a single white state out of the l games used in the verification protocol. Let us now consider l such games, and let $p^{(i)}_{\mathrm{Ver}_j}$ be the error probability for honest verifier j on the i'th run of the verification protocol. We claim that when we have l independent white states (in the sense that each x^i is chosen independently), it is still the case that

$$p^{(i)}_{\mathrm{Ver}_1} + p^{(i)}_{\mathrm{Ver}_2} \geq \frac{1}{2} - \frac{1}{2(n-1)} \tag{10.25}$$

for all i, regardless of the outcomes of previous measurements made by the verifiers. Though intuitively reasonable, this claim is far from trivial, but can be proved using a teleportation argument due to Croke and Kent [55] (See Appendix B) so that, essentially, we can imagine the adversary acts independently on each game in the verification protocol. Therefore, on each and every white state, at least one verifier must have an error probability of at least

$$\frac{1}{2}(p^{(i)}_{\mathrm{Ver}_1} + p^{(i)}_{\mathrm{Ver}_2}) = \frac{1}{4} - \frac{1}{4(n-1)}. \tag{10.26}$$

Overall, if we include the effects of r register manipulation and auxiliary verifications, at least one verifier, say Ver_1, must have an average error probability over all l games of at least

$$e_{\min} = \frac{997}{999}\left(\frac{1}{4} - \frac{1}{4(n-1)}\right) \approx \frac{1}{4} - \frac{1}{4(n-1)} \tag{10.27}$$

Using Hoeffding's inequality, the probability of both verifiers accepting the coin can

be bounded as

$$\begin{aligned}&\mathbb{P}(\text{Both Ver}_1 \text{ and Ver}_2 \text{ generate outcome “Valid”})\\&\quad\leq \mathbb{P}(\text{Ver}_1 \text{ generates outcome “Valid”})\\&\quad\leq e^{-2l\delta^2},\end{aligned} \tag{10.28}$$

where $\delta = (e_{\min} - \beta)/2$, as above. As long as $\beta < e_{\min}$, the Hoeffding bound can be used to show that it becomes exponentially unlikely for both verifiers to pass the verification protocol. By increasing the maximum noise tolerance of the protocol we increase the size of δ, thereby allowing smaller sample sizes in the verification protocol, which increases the re-usability of coins. If we choose $n = 4$, our scheme would be able to tolerate 16.6% noise, and for $n = 14$ it can tolerate up to 23% noise. This concludes the proof of security against forging.

In the next section, we prove an upper bound on the error tolerance achievable for a general class of classical verification quantum money schemes, and show this bound limits to 25% as the dimension of the underlying states is increased. This implies that our protocols are nearly optimal in terms of error tolerance. When proving this result, we assume only that the coin is a collection of quantum states each identified with a secret classical string, and that to verify the coin the holder must declare a number of single bit values which can be checked against the classical record.

10.5 Maximum achievable noise tolerance

Suppose we have a scheme in which the coin consists of many independently chosen n-dimensional pure quantum states, $\phi_x = |\phi_x\rangle\langle\phi_x|$, with $x \in X$ and where x is a classical bit string chosen according to some probability distribution. To verify each state, the holder performs some POVM, $\mathcal{M}_x = \{M_x^{\text{cor}}, M_x^{\text{inc}}\}$, to ascertain one bit of information about each of the states used in the verification protocol. The bit values resulting from the measurement outcomes are checked against a classical record to verify whether the coin is genuine or not.

Lemma 10.4. *For any quantum money scheme of the above type, the maximum tolerable noise, e_{max}, must be less than*

$$e_{max} \leq \frac{1}{2} - \frac{1}{4}\frac{n+2}{n+1}. \tag{10.29}$$

Proof. We prove this by explicitly illustrating a strategy available to the ad-

versary. The adversary holds the unknown state ϕ_x, which lives in Hilbert space $\mathcal{H}$. She extends the state to $\phi_x \otimes \Phi$, where $\Phi = \frac{1}{n}\mathbb{1}_n$, and symmetrises the system. Specifically, she performs the mapping

$$\phi_x \otimes \Phi \to S_2(\phi_x \otimes \Phi)S_2, \tag{10.30}$$

where S_2 is the projector onto $\mathcal{H}_+^2$, the symmetric subspace of $\mathcal{H}^{\otimes 2}$, and where the state on the right hand side is not normalised. The resulting normalised state of each clone is [184]

$$\eta_x = v\phi_x + (1-v)\Phi, \tag{10.31}$$

where $v := \frac{1}{2}\,\frac{n+2}{n+1}$. By the correctness requirement of quantum money schemes, an honest measurement on the correct state should always give a correct answer so that the coin is declared valid, i.e.

$$\mathrm{Tr}(M_x^{\mathrm{cor}}\phi_x) = 1. \tag{10.32}$$

We further assume that, without access to the state ϕ_x, the adversary has no information on x and can do no better than to guess randomly. This means her probability of declaring a correct bit value is $1/2$, i.e.[3]

$$\mathrm{Tr}(M_x^{\mathrm{cor}}\Phi) = 1/2. \tag{10.33}$$

Both honest verifiers hold the state η_x. Using Eqs. (10.32) and (10.33), the probability that an honest verifier gets a correct measurement outcome is

$$\begin{aligned}\mathrm{Tr}(M_x^{\mathrm{cor}}\eta_x) &= v\mathrm{Tr}(M_x^{\mathrm{cor}}\phi_x) + (1-v)\mathrm{Tr}(M_x^{\mathrm{cor}}\Phi)\\ &= v + \frac{(1-v)}{2}.\end{aligned} \tag{10.34}$$

Expressing v in terms of the dimension of the system shows that this strategy (which is always available to the adversary) leads to the honest verifiers finding an error

[3]Note that this assumption holds for all hidden matching quantum money schemes considered, and for any scheme in which the verification protocol involves declaring many single bit values which are later checked. Nevertheless, there may be protocols in which the verification protocol involves checking many m-bit outcomes, in which case the more reasonable assumption would be

$$\mathrm{Tr}(M_x^{\mathrm{cor}}\Phi) = 1/2^m.$$

To our knowledge such a scheme does not exist, but if higher error tolerance is desired our proof suggests looking into such schemes.

rate of

$$e_{\max} = \frac{1}{2} - \frac{1}{4}\frac{n+2}{n+1}, \tag{10.35}$$

and so for any such scheme to be secure an honest participant must expect an error rate less than $e_{\max}$ in an honest run of the protocol.

Our analysis shows that for any scheme with $n = 4$ the tolerable noise is at most 20%, which complements our results in Section 10.4.1 where we described a protocol with $n = 4$ which tolerated noise up to 16.6%. For $n = 14$, the bound in this section shows that any such scheme has a noise tolerance of at most 23.3%. For $n = 14$, our protocol can achieve an error tolerance of 23.03%, and so it is nearly optimal. As we increase the dimension of the quantum states used for the coins, the upper bound on the tolerable noise approaches 25% which coincides with our conjecture for the tolerable noise in our protocols above.

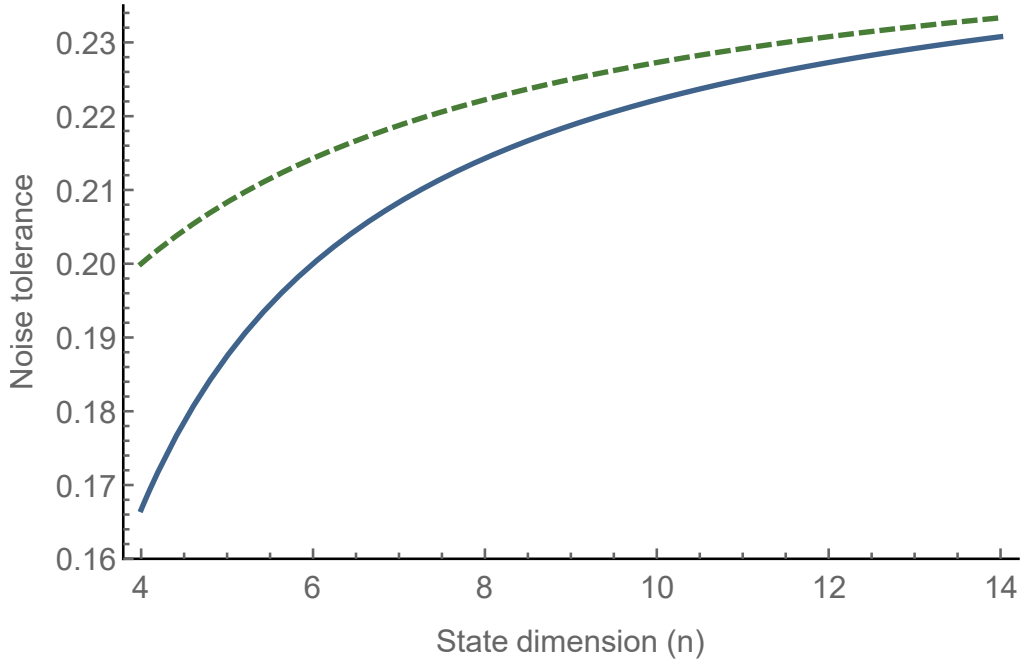


Figure 10.5: Plot showing the theoretical bound on protocol noise tolerance (dotted line) and the noise tolerance achieved by the protocols in Section 10.4 (bold line) as the dimension of the underlying systems increase.

10.6 Experimental implementation

The protocol presented in Section 10.4 gives rise to three main technical challenges when one considers experimental implementations, namely: the security analysis provided does not account for losses; the bank requires a source of complex, high-dimensional states; and the protocol requires that the coin holders have the ability to store states in quantum memory. In this section we address the first two issues so

that a proof-of-principle implementation of the verification algorithm of the quantum money schemes could be performed with current technology.

10.6.1 Detector losses

Here we tackle the first of the issues, and consider an implementation in which the verifiers use imperfect detectors with efficiency η. We assume that all detector losses are random and cannot be manipulated by the adversary. In this chapter we do not consider channel loss, as we assume that coin transfers occur over short distances, meaning channel losses are less relevant. Nevertheless, many of the methods presented here would remain valid in the presence of small channel loss with only minor modifications necessary. Note that detectors are employed by the holder and not the bank.

To incorporate detector loss, it is necessary to modify the verification protocol, previously stated in Section 10.4, so that it becomes

Ver Algorithm

1. The holder randomly chooses a subset of indices, $L \subset [q]$, with $l = |L|$, such that $r_i = 0$ for each $i \in |$. The indices $i \in L$ specify the selection of games G_i which will be used as tests for the verification procedure. For each $i \in L$, the holder then sets the corresponding bit of r to be 1 so that this game cannot be used in future verifications.

2. For each $i \in L$, the holder picks a relation σ_i' at random from $\mathcal{R}$ and applies the appropriate measurement to get answer d_i. If there is no measurement outcome we say the measurement was unsuccessful and set $d_i = \emptyset$. We define the number of successful measurement outcomes to be l'.

3. If $l' < l_{min} := (\eta - \epsilon)l$, where $\epsilon > 0$ is a small security parameter, the verifier aborts the protocol.

4. The holder sends all triplets (i, σ_i', d_i) to the bank.

5. The bank checks that $s < T$, where T is the pre-defined maximum number of allowed verifications for the coin. If $s = T$, the bank declares the coin as invalid.

6. For each i, the bank checks whether the answer is correct by comparing (i, σ_i', d_i) to the secret x^i values. The bank ignores those outcomes for which

$d_i = \emptyset$, and accepts the coin as valid only if more than $l'(c-\delta)$ of the answers are correct, where $c = 1-\beta$ is a measure of the channel correctness and δ is a small positive constant.

7. The bank updates s to $s+1$.

Correctness

Correctness of the scheme follows from Hoeffding's inequality. When all participants are honest, it is exponentially unlikely for l' to be less than l_{min}, so the protocol will not abort, except with a negligible probability. If the protocol does not abort, the verifier has at least l_{min} successful measurement outcomes, each with an independent probability c of being correct. Overall, the probability of the verification failing is bounded by

$$\mathbb{P}(\text{Ver fails}) \leq \exp\left[-2l_{min}\delta^2\right] + \exp[-2l\epsilon^2], \tag{10.36}$$

where now $\delta = (e'_{\min} - \beta)/2$, with $e'_{\min}$ derived in Eq. (10.40) below as the minimum average error rate achievable by the adversary.

Unforgeability

Since the protocol now includes detector losses, the adversary may not have to send states to each verifier for each game in the verification protocol, and she could attempt to hide losses arising from her strategy in the losses arising from detector inefficiency. As a consequence, the set of strategies available to the adversary is increased, and we must make sure our arguments in Section 10.4.1 still apply.

Let U_1 and U_2 be q-bit strings representing whether or not the adversary sent a state to Ver_1 and Ver_2 respectively, for each of the q games created by the bank. An entry of 1 means the adversary sent a state to the verifier, while an entry of 0 means the adversary did not send a state to the verifier. We want to show that, in order for the protocol not to abort, $W(U_i) \geq \gamma q$, where $\gamma := 1 - \frac{3\epsilon}{\eta}$ and W is the Hamming weight. Suppose $W(U_i) = \gamma q$. Then, in Step 1 of the verification protocol, Ver_i takes a sample, V_i, consisting of l of the entries of U_i. Hoeffding's inequality gives

$$P\Big(W(V_i) \leq (\gamma + \frac{\epsilon}{\eta})l\Big) \geq 1 - \exp[-2\frac{\epsilon^2}{\eta^2}l]. \tag{10.37}$$

If $W(V_i) \leq (\gamma + \frac{\epsilon}{\eta})l$, then the probability of at least l_{min} successful measurement

outcomes is given by

$$P\Big(\text{At least } l_{min} \text{ succ. meas. } \mid W(V_i) \leq (\gamma + \frac{\epsilon}{\eta})l\Big) \leq \exp[-2l\epsilon^2]. \tag{10.38}$$

The probability of the protocol proceeding past Step 3 of verification is therefore

$$P(\text{No Abort}|W(U_i) = \gamma q) \leq \exp[-2\frac{\epsilon^2}{\eta^2}l] + \exp[-2\epsilon^2 l]. \tag{10.39}$$

In what follows we assume $W(U_i) \geq \gamma q$, since otherwise the above shows that the verifiers will abort with near certainty. This means the adversary is able to use any strategy that leads to channel losses of at most $\frac{3\epsilon}{\eta}$ for each verifier, as these can be hidden within the normal fluctuations of detector loss. Suppose there is a strategy which gives at least $(1 - \frac{3\epsilon}{\eta})q$ states to each verifier, and which leads to an average error probability (on only the states tested) of $e'_{\min}$ for at least one of the verifiers. Then, there is a strategy which gives q states to each verifier, and leads to an average error probability for at least one of the verifiers of $(1 - \frac{3\epsilon}{\eta})e'_{\min} + \frac{3\epsilon}{2\eta}$ (the adversary simply sends the maximally mixed state to each verifier in place of the $\frac{3\epsilon}{\eta}$ losses). Since this strategy falls under the scope of the analysis in Section 10.4.1, we know that the resulting error rate must be at least $e_{\min}$, which means

$$e'_{\min} \geq \frac{e_{\min} - \frac{3\epsilon}{2\eta}}{1 - \frac{3\epsilon}{\eta}}. \tag{10.40}$$

The parameter ϵ can be chosen to be arbitrarily small by increasing the sample size l. As such, the protocol is able to handle arbitrarily large detector losses, and leads to noise tolerance that can be kept arbitrarily close to the noise tolerance derived for the case of perfect detectors.

Each verifier tests at least l_{min} states, and at least one verifier expects an error rate of $e'_{\min}$. The probability of this verifier passing the test is bounded as

$$P(\text{Error rate} < e'_{\min} - \delta) \leq \exp[-2l_{min}\delta^2]. \tag{10.41}$$

Combining Eqs. (10.39) and (10.41), the probability that the adversary is able to forge a coin is given by

$$P(\text{Forgery}) \leq \exp[-2\frac{\epsilon^2}{\eta^2}l] + \exp[-2l\epsilon^2] + \exp[-2l_{min}\delta^2] \tag{10.42}$$

10.6.2 Coherent state implementation

In this section we tackle the second issue arising when considering experimental realisations of the scheme – the bank must create hidden matching states of the form in Eq. (10.1), which are high-dimensional states of high complexity. The implementation of hidden matching quantum retrieval games has been studied extensively in Ref. [183], where the coherent state mapping defined in Ref. [185] was used to approximate each hidden matching state by a sequence of n coherent states of the form

$$|\alpha, x\rangle = e^{-\frac{|\alpha|^2}{2}} \sum_{k=0}^{\infty} \frac{\alpha^k}{k!} (a_x^\dagger)^n |0\rangle = \bigotimes_{i=1}^{n} \left| (-1)^{x_i} \frac{\alpha}{\sqrt{n}} \right\rangle, \tag{10.43}$$

where

$$a_x^\dagger = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} (-1)^{x_i} b_i^\dagger \tag{10.44}$$

and $\{b_1^\dagger, b_2^\dagger, \ldots, b_n^\dagger\}$ are the creation operators of the n modes. We call each sequence of coherent states a block, so that a single block is used to approximate a hidden matching state. As outlined in Ref. [183], Bob's measurement can then be performed using linear optics circuits and single-photon detectors.

In the absence of a phase reference, the phase of each block is randomised, which implies that each block is equivalent to a classical mixture of number states [91]. More specifically, writing $\alpha = e^{i\theta}|\alpha|$, we have

$$\int_0^{2\pi} \frac{d\theta}{2\pi} |\alpha, x\rangle\langle\alpha, x| = e^{-|\alpha|^2} \sum_{k=0}^{\infty} \frac{|\alpha|^{2k}}{k!} |k\rangle\langle k|_x, \tag{10.45}$$

where $|k\rangle\langle k|_x$ is a state of k photons in the mode $a_x^\dagger$. Thus, the probability of obtaining a particular number of photons depends only on α, which is a free parameter within the coherent state mapping. We consider the following three cases:

Zero photons in the block

In this case the state emitted is simply the vacuum state. If the adversary chooses to forward a state on to the verifiers, she can do no better than to induce a 50% error rate, and it is simple to show that it is never beneficial for her to do so. This scenario can therefore be considered a "source" loss, as opposed to a channel or detector loss. Crucially, since these losses are not controllable by the adversary, they can be treated in the same manner as detector losses in Section 10.6.1 simply by including the source loss into the detector loss parameter, η. The probability of zero photons being emitted is $p_0 = e^{-|\alpha|^2}$.

One photon in the block

In this case, the state emitted is equivalent to the ideal hidden matching state in Eq. (10.1) since

$$|1\rangle_x = a_x^\dagger |0\rangle = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} b_i^\dagger |0\rangle = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} (-1)^{x_i} |i\rangle , \tag{10.46}$$

where $|i\rangle$ is a single-photon state in the mode b_i. Therefore, whenever the bank's source emits a single-photon, the analysis in Section 10.4.1 applies. The probability of one photon being emitted is $p_1 = |\alpha|^2 e^{-|\alpha|^2}$.

More than one photon in the block

In this case we assume the worst case scenario: whenever the source emits more than one photon to represent a hidden matching state, the adversary can perfectly forge that state. The resulting error rate for the adversary is $e'_{\min}(\frac{p_1}{p_1+p_{2+}})$, where $p_{2+} = 1 - p_0 - p_1$. For small $|\alpha|$, $p_{2+} \approx \frac{|\alpha|^4}{2}$, while $p_1 \approx |\alpha|^2$, so that $p_{2+} \ll p_1$ and the adversary's error probability is almost unchanged by using coherent states.

10.7 Conclusion

We presented a family of unconditionally secure classical verification quantum money schemes which are tolerant to noise up to 23%, and which we conjecture tolerate noise up to 25%. We further proved that 25% is the maximum noise tolerance achievable for a wide class of quantum money schemes, including all classical verification secret-key schemes previously proposed. The security of our schemes depends on the difference between maximum tolerable noise and expected noise, meaning the increase in maximum tolerable noise increases the efficiency of our scheme, allowing for smaller, more re-usable coins. The techniques we use to prove security differ considerably to previous papers, and the re-usability of our coins is optimal [174] in that it scales linearly with the number of qubits in the coin. This is a significant improvement when compared to Ref. [177], in which the re-usability scales as $q^{1/3}$, and Ref. [174], in which re-usability scales as $q^{1/4}$, where q is the total number of qubits in the coin. With realistic assumptions on experimental equipment, we expect that, using $n = 8$, a coin containing 10^9 qubits would use $l = 18,000$ states for each verification, and would be re-usable $T = 100$ times for a security level of 10^{-6}. Lastly, we suggested methods of adapting our techniques to facilitate experimental

implementations of the scheme. We show that the schemes can be implemented using weak coherent states even in the presence of limited detector efficiency.

Chapter 11

Conclusion

For thousands of years signatures have fulfilled an essential role in safeguarding the integrity, authenticity and transferability of communications. With the explosion of information technologies, the importance and prevalence of signatures has increased tremendously, and it is hard to imagine a future in which some form of signature is not used to secure communications. As time progresses, inevitable technological advances result in schemes that must provide security against ever more powerful adversaries if they are to remain useful. Quantum computers are particularly relevant to the present day, since they threaten to render digital signature schemes such as RSA, DSA and ECDSA obsolete.

In this thesis we have focused on USS schemes – signature schemes designed to provide security against even the most powerful adversary. We have looked at both quantum USS schemes, in which security guarantees are derived from the laws of quantum mechanics, and classical USS schemes, who's security relies only on mathematical arguments. The cost of such a high security level is that USS schemes are much less efficient than signature schemes providing lower levels of security, and require a set-up phase to distribute secret key amongst all protocol participants. The latter requirement means that USS schemes will not be a suitable replacement for many core applications of digital signatures, but should instead be viewed as a complement to existing QKD networks. In such networks, high security is clearly valued and each node already has the ability to generate and distribute a secret key.

In Chapter 6 we described and analysed the first quantum USS scheme that is both unconditionally secure and experimentally realisable. The scheme was more efficient than previous quantum USS schemes, and benefitted from many similarities to QKD making it cheap and simple to implement in existing QKD networks. Interestingly, we also found that the scheme could be performed over channels too noisy

for QKD. The scheme was then extended in Chapter 7 to make it measurement-device-independent, thereby adding a further layer of real-world security.

Quantum mechanics opens up vast new possibilities for cryptographic and communication technologies. However, its use is often expensive and leads to additional complexities that make experimental implementations difficult or impossible with current technology. Therefore, it is both interesting and important to ascertain exactly what advantages quantum mechanics provides for a given task. In Chapter 8 we presented a classical USS scheme, the hash scheme, which enjoys all of the benefits of quantum USS schemes as well as being hugely more efficient. In fact, the scheme is so efficient that it could realistically be used to sign real-world data. The scheme extends classical authentication techniques to also provide transferability. Since QKD already uses classical authentication as a sub-protocol, this again means that the scheme can be easily implemented in an existing QKD network, requiring no new hardware and only minimal software modifications.

Unfortunately for quantum USS schemes, the hash scheme means that there are no known advantages in directly using quantum mechanics to construct USS schemes. Nevertheless, all USS schemes are reliant on unconditionally secure key distribution, and they provide a set of useful security guarantees essential to many communications. As such, all USS schemes should be thought of as an excellent application of the quantum technology QKD.

In Chapters 9 and 10 we departed from the direct study of USS schemes, and instead explored two related quantum protocols – oblivious transfer and quantum money. Motivated by the close connection between USS schemes and oblivious transfer, we extended the bounds on what is known to be impossible in stand-alone 1-out-of-2 oblivious transfer. Due to the importance of oblivious transfer in multiparty computations, we believe the resulting bounds are interesting, and hope that they will help to shed light on the potential applications of imperfect oblivious transfer.

Lastly, in Chapter 10 we described and analysed a new secret-key quantum money scheme that is more error-tolerant than all previous schemes. We further showed that the error-tolerance achieved is essentially optimal for a wide class of secret-key quantum money schemes. Continuing the theme of searching for practical quantum protocols, we described methods by which our scheme can be reformulated into one which is both secure and experimentally implementable. This paves the way for the first experimental demonstration of quantum coin creation, transmission and verification.

In the years to come, quantum mechanics will undoubtedly continue to play a central role in the fields of communications, computing and cryptography. The advantages offered by quantum technologies over purely classical ones is remarkable, but these additional powers are not gained without cost. Many existing cryptographic protocols designed to protect important services will need to be updated to provide resilience against powerful quantum adversaries. This thesis has focused on one such protocol, that of signing information. Overall, we hope that the results contained in this thesis have helped to explore, consolidate and clarify the role and potential applications of USS schemes in the post-quantum era.

Appendix A

A.1 Finite-size estimates

In order to calculate the min-entropy in Eq. (6.2), we must estimate the three quantities $s^-_{X,0}$, $s^-_{X,1}$ and $\phi^+_{X,1}$. The method used to estimate these quantities is described in Ref. [114]. For completeness, we provide an overview of their arguments here.

Recall that $s^-_{X,0}$ and $s^-_{X,1}$ are estimates of the number of counts (sent and measured in the X basis) containing zero and one photon respectively. $\phi^+_{X,1}$ is an estimate of the phase error rate in the X basis counts coming from single-photon pulses. Unfortunately, these quantities are not directly observable, and as such the aim of this section is to show how they can be estimated using observed statistics.

Recall that the X basis raw key is generated by randomly selecting a sample of bits from the total of all X basis counts collected. Across all X basis counts, the exact number of counts corresponding to each intensity level is known, and from this the expected number of each intensity level going into the raw key can be derived. In the asymptotic limit of infinitely many X basis counts in the raw key, the true number of counts at each intensity level will tend to the expected number of counts at each intensity level. As such, we can lower bound $s_{X,0}$ as

$$s_{X,0} \geq \frac{\tau_0}{u_2 - u_3}\left(\frac{u_2 e^{u_3} n^*_{X,u_3}}{p_{u_3}} - \frac{u_3 e^{u_2} n^*_{X,u_2}}{p_{u_2}}\right), \tag{A.1}$$

where n^*_{X,u_i} is the expected number of X basis counts coming from pulses with intensity u_i, and $\tau_n := \sum_{u_i} p_{u_i} e^{-u_i} u_k^n / n!$. In the finite setting the true number of counts at each intensity level, n_{X,u_i}, cannot be set to the expected value. Nevertheless we are able to bound n_{X,u_i} from above and below with high probability. Specifically, if

the raw key contains $L+k$ counts, Hoeffding's inequalities [58] give

$$\begin{aligned} n^-_{X,u_i} &:= n^*_{X,u_i} - \delta(L+k, \epsilon_{PE}) \leq n_{X,u_i} \\ n^+_{X,u_i} &:= n^*_{X,u_i} + \delta(L+k, \epsilon_{PE}) \geq n_{X,u_i}. \end{aligned} \tag{A.2}$$

These bounds each hold with probability at least $1-\epsilon_{PE}$, where $\delta(L+k, \epsilon_{PE}) := \sqrt{(L+k)\ln(1/\epsilon_{PE})/2}$. Replacing the n^*_{X,u_i} in Eq. (A.1) by the corresponding worst-case finite-size estimate leads to a finite-size lower bound on $s_{X,0}$, which we call $s^-_{X,0}$, and which holds with probability at least $1-2\epsilon_{PE}$.

Similarly, we can bound $s^-_{X,1}$ as

$$\begin{aligned} s^-_{X,1} \geq & \frac{u_1\tau_1}{u_1(u_2-u_3)-(u_2^2-u_3^2)}\left[\frac{e^{u_2}n^-_{X,u_2}}{p_{u_2}} - \frac{e^{u_3}n^+_{X,u_3}}{p_{u_3}}\right. \\ & \left. + \frac{u_2^2-u_3^2}{u_1^2}\left(\frac{s^-_{X,0}}{\tau_0} - \frac{e^{u_1}n^+_{X,u_1}}{p_{u_1}}\right)\right]. \end{aligned} \tag{A.3}$$

The X basis phase errors are not directly observed in the protocol. Instead, we relate $\phi^+_{X,1}$ to the bit error rate in the Z basis. As in Appendix B of [114], we have

$$\phi^+_{X,1} \leq \frac{v^+_{Z,1}}{s^-_{Z,1}} + \gamma\left(\alpha_1, \frac{v^+_{Z,1}}{s^-_{Z,1}}, s^-_{Z,1}, s^-_{X,1}\right), \tag{A.4}$$

where α_1 is such that $0<\alpha_1<\epsilon$, ϵ is the smoothing parameter in the smooth min-entropy, $v^+_{Z,1}$ is the upper bound on the number of errors in Z basis counts coming from single photon pulses, and

$$\gamma(a,b,c,d) := \sqrt{\frac{(c+d)(1-b)b}{cd\ln 2}\log\left[\frac{c+d}{cd(1-b)b}\frac{1}{a^2}\right]}. \tag{A.5}$$

All quantities on the right hand side of Eq. (A.4) are known, except $v^+_{Z,1}$ which we can find as

$$v^+_{Z,1} \leq \frac{\tau_1}{u_2-u_3}\left(\frac{e^{u_2}m^+_{Z,u_2}}{p_{u_2}} - \frac{e^{u_3}m^-_{Z,u_3}}{p_{u_3}}\right), \tag{A.6}$$

where the $m^\pm_{Z,u_i}$ are the upper and lower bounds on the true number of bit errors coming from Z basis counts of intensity u_i. These quantities are found similarly to Eq. (A.2).

A.2 Proofs of Lemmas 6.2 and 6.3

Proof of Lemma 6.2. To prove the lemma we will show that for *any* $\overline{\tau}'_{XF} \in B^\epsilon(\tau_{XF})$, and any sub-normalised σ'_F, there exists a classical $\overline{\tau}_{XF} \in B^\epsilon(\tau_{XF})$ and sub-normalised σ_F such that

$$H_{\min}(\overline{\tau}_{XF}|\sigma_F) \geq H_{\min}(\overline{\tau}'_{XF}|\sigma'_F). \tag{A.7}$$

Since the smooth min-entropy $H^\epsilon_{\min}(X|F)_\tau$ involves a maximisation over all states ϵ-close to τ_{XF} (see Eqs. 3.16 and 3.18) the result then follows.

For any $\overline{\tau}'_{XF} \in B^\epsilon(\tau_{XF})$, choose $\overline{\tau}_{XF} := \mathcal{E}_{XF}(\overline{\tau}'_{XF})$, where $\mathcal{E}_{XF}$ denotes the projection onto the $\{|x\rangle\,|f\rangle\}_{x,f}$ basis. We first show that $\overline{\tau}_{XF} \in B^\epsilon(\tau_{XF})$, and then show Eq. (A.7). Since τ_{XF} is classical in the $\{|x\rangle\,|f\rangle\}_{x,f}$ basis, $\mathcal{E}_{XF}(\tau_{XF}) = \tau_{XF}$. Therefore,

$$P(\overline{\tau}_{XF}, \tau_{XF}) = P\big(\mathcal{E}_{XF}(\overline{\tau}'_{XF}), \mathcal{E}_{XF}(\tau_{XF})\big) \leq P(\overline{\tau}'_{XF}, \tau_{XF}) \leq \epsilon, \tag{A.8}$$

where the first inequality follows from the monotonicity of the purified distance, and the second inequality follows because $\overline{\tau}'_{XF} \in B^\epsilon(\tau_{XF})$. This shows that $\overline{\tau}_{XF} \in B^\epsilon(\tau_{XF})$.

To prove Eq. (A.7), recall Definition 3.8 which says

$$H_{\min}(\overline{\tau}'_{XF}|\sigma'_F) := \sup\{\lambda \in \mathbb{R} \,:\, \overline{\tau}'_{XF} \leq 2^{-\lambda}\mathbb{1}_X \otimes \sigma'_F\}. \tag{A.9}$$

We $\sigma_F := \mathcal{E}_F(\sigma'_F)$, where $\mathcal{F}$ is the projection onto the $\{|f\rangle\}$ basis. Applying $\mathcal{E}_{XF}$ to both sides of $\overline{\tau}'_{XF} \leq 2^{-\lambda}\mathbb{1}_X \otimes \sigma'_F$ gives

$$2^{-\lambda}\mathbb{1}_X \otimes \sigma'_F - \overline{\tau}'_{XF} \geq 0 \Rightarrow 2^{-\lambda}\mathbb{1}_X \otimes \sigma_F - \overline{\tau}_{XF} \geq 0. \tag{A.10}$$

Equivalently, Eq. (A.10) shows that $H_{\min}(\overline{\tau}_{XF}|\sigma_F) \geq H_{\min}(\overline{\tau}'_{XF}|\sigma'_F)$, from which the result follows. □

Proof of Lemma 6.3. Let $\mathcal{X}$ be the set of all n-bit strings and let $S^r_x := \{x' \in \mathcal{X} : d(x, x') \leq r\}$, where d is the Hamming distance. When using F to guess X, Eve's average probability of making fewer than r mistakes is at most

$$q_r = \sum_f Q_F(f) \max_{\tilde{x}} \sum_{x' \in S^r_{\tilde{x}}} Q_{X|F=f}(x'), \tag{A.11}$$

where Q_F is the marginal distribution of Q_{XF}. This can be understood as follows. Eve is successful in making fewer than r errors if Eve guesses $\tilde{x}$, and $X = x^*$ such that $x^* \in S^r_{\tilde{x}}$. In other words, given Eve's guess is $\tilde{x}$, she is successful if the event $\mathcal{E}_{\tilde{x}} = \{X = x^* \ : \ x^* \in S^r_{\tilde{x}}\}$ occurs. Therefore, for each fixed $F = f$, Eve's optimal strategy is to guess the value $\tilde{x}$ for which the probability of $\mathcal{E}_{\tilde{x}|f}$ occuring is maximal. The conditional probability that $x^* \in S^r_{\tilde{x}}$, given $F = f$, can be written as $\mathbb{P}(\mathcal{E}_{\tilde{x}}|F = f) = \sum_{x' \in S^r_{\tilde{x}}} Q_{X|F=f}(x')$, hence Eq. (A.11). Continuing, we have

$$\begin{aligned} q_r &= \sum_f Q_F(f) \max_{\tilde{x}} \sum_{x' \in S^r_{\tilde{x}}} Q_{X|F=f}(x') \\ &\leq \sum_f Q_F(f) \sum_{x' \in S^r_{\tilde{x}}} \max_x Q_{X|F=f}(x) \\ &= b_n^r \sum_f Q_F(f) \max_x Q_{X|F=f}(x) \\ &= b_n^r 2^{-H_{\min}(X|F)}, \end{aligned} \tag{A.12}$$

where the second equality uses $|S^r_{\tilde{x}}| = b^r_n$, and the final equality uses the that fact that, on classical states, $H_{\min}(X|F) = -\log_2 \sum_f Q_F(f) \max_x Q_{X|F=f}(x)$, as in Ref. [80]. This proves the lemma. □

Appendix B

Overview of teleportation strategy

In Section 10.4.1 we claimed that the adversary cannot use coherent attacks on multiple states in order to beat the bound given in Eq. (10.24), even when conditioned on the states chosen by the bank, and on the outcomes of previous measurement results found by the verifiers. In this section we formally prove our claim using a teleportation argument similar to the one introduced by Croke and Kent in Ref. [55], so that each game can essentially be viewed as independent of all others.

In order to apply the teleportation argument, we must first introduce a modified individual setting, in which the adversary is allowed an additional ability. We show that this modification does not help the adversary to cheat. We then show that any coherent strategy can be transformed into a modified individual strategy. Therefore, any coherent strategy cannot beat the bounds proved for the unmodified individual case, as claimed.

Modified individual attacks

In the individual setting, the verifiers each receive a single hidden matching state and apply the verification protocol to test its authenticity. As specified by the protocol, the verifiers randomly choose to measure the state they receive using one of the matching measurements. We include this random choice of matching into the mathematical description of the measurement, and group the outcomes to be either "correct" or "incorrect". It can be shown that if the bank creates $\phi_x = |\phi_x\rangle \langle\phi_x|$, the verifiers measurement is described by the POVM

$$\Gamma_x = \{\Gamma^{\mathrm{cor},x}, \Gamma^{\mathrm{inc},x}\} = \frac{n}{2(n-1)} \left\{ \frac{n-2}{n}\mathbb{I} + \phi_x,\, \mathbb{I} - \phi_x \right\}. \tag{B.1}$$

Suppose now the adversary has the additional power of being able to force the verifiers to apply a correction unitary (which will be the teleportation corrections)

to their measurement outcomes before they are sent to the bank. The adversary must specify the correction operation before sending the states to the verifiers, and, crucially, the correction operation is such that it is simply a permutation of the set of hidden matching states. For example, suppose the teleportation operation takes input $|\phi_x\rangle$ and outputs $|\phi_{x'}\rangle$, with correction operator C. In this case, before sending the states, the adversary will tell the verifiers that they must apply correction C to their measurement outcomes. In effect then, the verifiers will measure

$$\Gamma_{x'} = \{\Gamma^{\text{cor},x'}, \Gamma^{\text{inc},x'}\} = \frac{n}{2(n-1)} \left\{ \frac{n-2}{n}\mathbb{I} + \phi_{x'},\ \mathbb{I} - \phi_{x'} \right\}, \tag{B.2}$$

since the correction applied to $\Gamma^{\text{inc},x'}$ is $\Gamma^{\text{inc},x}$. On average, given ϕ_x, it is not possible for the adversary to create two states, η_x and τ_x, such that $\text{Tr}[\Gamma^{\text{inc},x'}(\eta_x + \tau_x)] < p$, where $p := p_{\text{Ver}_1} + p_{\text{Ver}_2}$. If it were possible, then it would imply that the adversary can clone $\phi_{x'}$ better than what is allowed by quantum mechanics (and our arguments in Section 10.4.1). This is because if the adversary was given $\phi_{x'}$ he could easily transform it to ϕ_x by applying C, and then perform the strategy to get two copies with a fidelity higher than the bound proved in Section 10.4.1. Therefore the additional power given to the adversary does not allow her to decrease the value of $p_{\text{Ver}_1} + p_{\text{Ver}_2}$.

Coherent strategy

We now consider the case of N games created by the bank. The bank creates

$$\frac{1}{2^{Nn}} \sum_{x_1,x_2} |x_1\rangle\langle x_1|_{X_1} \otimes |x_2\rangle\langle x_2|_{X_2} \otimes |\phi_{x_1}\rangle\langle\phi_{x_1}|_A \otimes |\phi_{x_2}\rangle\langle\phi_{x_2}|_B . \tag{B.3}$$

The X_1 and A registers contain the first $N-1$ secret strings selected by the bank and the corresponding hidden matching states, respectively. The X_2 and B registers contain the N'th secret string selected by the bank and its corresponding hidden matching state. Only the A and B registers are accessible to the adversary. We assume for a contradiction that there exists a strategy available to the adversary such that, conditional on having obtained specific values in

1. The X_1 register, and

2. The verifiers' outcomes in previous measurements,

then the value of $p_{\text{Ver}_1} + p_{\text{Ver}_2}$ in the N'th game is decreased below the bound in Eq. (10.24).

We describe this strategy as follows – upon receiving the states from the bank, the adversary applies the unitary operation S_{ABC} so that the state becomes

$$\begin{aligned}
&\frac{1}{2^{Nn}} \sum_{x_1,x_2} |x_1\rangle\langle x_1|_{X_1} \otimes |x_2\rangle\langle x_2|_{X_2} \\
&\qquad\qquad \otimes S_{ABC}\Big(|\phi_{x_1}\rangle\langle\phi_{x_1}|_A \otimes |\phi_{x_2}\rangle\langle\phi_{x_2}|_B \otimes |0\rangle\langle 0|_C \Big) S_{ABC}^\dagger \qquad \text{(B.4)} \\
&= \frac{1}{2^{Nn}} \sum_{x_1,x_2} |x_1\rangle\langle x_1|_{X_1} \otimes |x_2\rangle\langle x_2|_{X_2} \otimes |\Psi^{x_1x_2}\rangle\langle\Psi^{x_1x_2}|_{AA'BB'C'} .
\end{aligned}$$

The A, A' registers are the spaces that contain the states that will be sent to Ver_1 and Ver_2 (resp.) for the first $N-1$ games. The B, B' registers are the spaces that contain the states that will be sent to Ver_1 and Ver_2 (resp.) for the N'th game. The C registers are auxiliary registers held by the adversary. We assume that the bank measures the X_1 register, and gets a state, x_1, which satisfies condition (1) of the strategy. The state held by the adversary is then

$$\frac{1}{2^n} \sum_{x_2} |\Psi^{x_1x_2}\rangle\langle\Psi^{x_1x_2}| . \qquad \text{(B.5)}$$

The adversary gives the A, A', B, B' parts of the state to the verifiers. The honest verifiers will first make measurements on systems A, A' and a possible post measurement state is

$$\frac{1}{2^n} \sum_{x_2} a_{x_1x_2} \Pi_{AA'} |\Psi^{x_1x_2}\rangle\langle\Psi^{x_1x_2}| \Pi_{AA'}^\dagger . \qquad \text{(B.6)}$$

We assume that $\Pi_{AA'}$ is a measurement outcome satisfying condition (2) of the strategy, so that the error probabilities on the N'th game are decreased. Here $a_{x_1x_2}$ is the normalisation term, $a_{x_1x_2} = 1/\mathrm{Tr}\big[\Pi_{AA'} |\Psi^{x_1x_2}\rangle\langle\Psi^{x_1x_2}| \Pi_{AA'}^\dagger\big]$.

The verifiers now each measure Γ_{x_2}, as defined in Eq. (B.2), on their B system. By assumption, the strategy then gives

$$\begin{aligned}
\frac{1}{2^n} \sum_{x_2} \Bigg[& a_{x_1x_2} \mathrm{Tr}\Big[\Gamma_B^{\mathrm{inc},x_2} \Pi_{AA'} |\Psi^{x_1x_2}\rangle\langle\Psi^{x_1x_2}| \Pi_{AA'}^\dagger\Big] \\
& + a_{x_1x_2} \mathrm{Tr}\Big[\Gamma_{B'}^{\mathrm{inc},x_2} \Pi_{AA'} |\Psi^{x_1x_2}\rangle\langle\Psi^{x_1x_2}| \Pi_{AA'}^\dagger\Big]\Bigg] < p.
\end{aligned} \qquad \text{(B.7)}$$

We now aim to prove that this leads to a contradiction.

Teleportation strategy

Supposing the above strategy exists, we explore what this enables the adversary to do in the individual case in the hopes of finding a contradiction. We suppose the bank creates

$$\frac{1}{2^n}\sum_{x_2}|x_2\rangle\langle x_2|_{X_2}\otimes|\phi_{x_2}\rangle\langle\phi_{x_2}|_B \tag{B.8}$$

and sends the B part to the adversary. The adversary can simulate the above strategy locally, by creating $|x_1\rangle$, $|\phi_{x_1}\rangle$ and the maximally mixed state on n dimensions $|\Phi\rangle$. After relabelling the registers, the adversary holds the state

$$\begin{aligned}\frac{1}{2^n}\sum_{x_2}|x_1\rangle\langle x_1|_{X_1}\otimes|x_2\rangle\langle x_2|_{X_2}\otimes|\phi_{x_1}\rangle\langle\phi_{x_1}|_A \\ \otimes|\phi_{x_2}\rangle\langle\phi_{x_2}|_D\otimes|0\rangle\langle 0|_C\otimes|\Phi\rangle\langle\Phi|_{BE}\,.\end{aligned} \tag{B.9}$$

To simulate the strategy in the previous section, the adversary applies S to the A, B and C registers, followed by a measurement on the resulting A, A' registers. Conditional on measurement outcome $\Pi_{AA'}$, she then applies a generalised Bell measurement on the D and E registers in order to teleport the unknown state $|\phi_{x_2}\rangle$ into the B register which was acted on by S (modulo a teleportation correction). If the appropriate measurement outcome is not found, the adversary does not perform the Bell measurement and instead starts again. The resulting state is

$$\frac{1}{2^n}\sum_{x_2}a_{x_1x_2'}\Pi_{AA'}\left|\Psi^{x_1x_2'}\right\rangle\left\langle\Psi^{x_1x_2'}\right|\Pi^\dagger_{AA'}. \tag{B.10}$$

Notice the state contains x_2' since the Bell measurement does not faithfully teleport the state, and a correction is required which we have not performed. If the dimension of the hidden matching states is a power of two, the correction operators are simply tensor products of the Pauli operators [186]. Crucially, all corrections define a bijective mapping between x_2' and x_2, so that as x_2 cycles over all possible values so does x_2', and the probabilities are not affected (all corrections are equally likely, which must be the case so that information is not communicated faster than light).

The state in Eq. (B.10) is the same as the state in Eq. (B.6), but the measurements applied by the verifiers are correlated with the X_2 register held by the bank. Therefore, the verifiers failure probabilities are not the same when measuring the

two states. Measurements on the state in Eq. (B.6) leads to a failure probability of

$$\frac{1}{2^n}\sum_{x_2}\Bigg[a_{x_1x_2}\mathrm{Tr}\Big[\Gamma_B^{\mathrm{inc},x_2}\,\Pi_{AA'}\left|\Psi^{x_1x_2}\right\rangle\left\langle\Psi^{x_1x_2}\right|\Pi_{AA'}^\dagger\Big] + a_{x_1x_2}\mathrm{Tr}\Big[\Gamma_{B'}^{\mathrm{inc},x_2}\,\Pi_{AA'}\left|\Psi^{x_1x_2}\right\rangle\left\langle\Psi^{x_1x_2}\right|\Pi_{AA'}^\dagger\Big]\Bigg], \tag{B.11}$$

while measurements on the state in Eq. (B.10) lead to a failure probability of

$$\frac{1}{2^n}\sum_{x_2}\Bigg[a_{x_1x_2'}\mathrm{Tr}\Big[\Gamma_B^{\mathrm{inc},x_2}\,\Pi_{AA'}\left|\Psi^{x_1x_2'}\right\rangle\left\langle\Psi^{x_1x_2'}\right|\Pi_{AA'}^\dagger\Big] + a_{x_1x_2'}\mathrm{Tr}\Big[\Gamma_{B'}^{\mathrm{inc},x_2}\,\Pi_{AA'}\left|\Psi^{x_1x_2'}\right\rangle\left\langle\Psi^{x_1x_2'}\right|\Pi_{AA'}^\dagger\Big]\Bigg]. \tag{B.12}$$

The difference being the appearance of x_2' in the second expression. Nevertheless, the two can be made equal if the verifiers are forced to apply the teleportation correction unitary to their measurement outcomes. In effect, this correction relabels the measurement outcomes so that $\Gamma^{\mathrm{inc},x_2} \to \Gamma^{\mathrm{inc},x_2'}$. Following this correction, the two expressions (B.11) and (B.12) are equal. This shows that the assumption in Eq. (B.7) leads to a contradiction, since it shows an individual attack in the modified scenario can achieve an error probability lower than p, but we know that the error probabilities achievable in the modified individual scenario are the same as for the unmodified individual scenario, hence the contradiction with our results in Section 10.4.1.

References

[1] Petros Wallden et al. "Quantum digital signatures with quantum-key-distribution components". *Physical Review A* 91 (2015), p. 042304.

[2] Werner Heisenberg. "Über den anschaulichen Inhalt der quantentheoretischen Kinematik und Mechanik". *Zeitschrift für Physik* 43.3-4 (1927), pp. 172–198.

[3] Howard Percy Robertson. "The uncertainty principle". *Physical Review* 34.1 (1929), p. 163.

[4] Albert Einstein, Boris Podolsky, and Nathan Rosen. "Can quantum-mechanical description of physical reality be considered complete?" *Physical Review* 47.10 (1935), p. 777.

[5] John Stewart Bell. *Speakable and unspeakable in quantum mechanics: Collected papers on quantum philosophy*. Cambridge university press, 2004.

[6] Claude Elwood Shannon. "A mathematical theory of communication". *ACM SIGMOBILE Mobile Computing and Communications Review* 5.1 (2001), pp. 3–55.

[7] C Shannon. "Communication theory of secrecy systems". *Bell syst. Tech. J.* 28 (1949), pp. 656–715.

[8] Whitfield Diffie and Martin E Hellman. "New directions in cryptography". *IEEE Trans. Inform. Theory* 22.6 (1976), pp. 644–654.

[9] Ronald L Rivest. *Cryptography, Handbook of theoretical computer science (vol. A): algorithms and complexity*. 1991.

[10] Charles H Bennett and Gilles Brassard. "Quantum cryptography: Public key distribution and coin tossing". New York, USA: IEEE, 1984, pp. 175–179.

[11] Charles H. Bennett et al. "Quantum cryptography, or unforgeable subway tokens". *Advances in Cryptology: Proceedings of Crypto '82, Santa Barbara*. Berlin: Plenum Press, 1983, pp. 267–275.

[12] Ronald L Rivest, Adi Shamir, and Len Adleman. "A method for obtaining digital signatures and public-key cryptosystems". *Communications of the ACM* 21.2 (1978), pp. 120–126.

[13] Taher ElGamal. "A public key cryptosystem and a signature scheme based on discrete logarithms". *Proceedings of CRYPTO 84 on Advances in Cryptology, LNCS, Santa Barbara, USA, 1984*. Vol. 196. Berlin, Heidelberg: Springer, 1985, pp. 10–18. ISBN: 3540156585.

[14] Don Johnson, Alfred Menezes, and Scott Vanstone. "The elliptic curve digital signature algorithm (ECDSA)". *Int. J. Inf. Sec.* 1.1 (2001), pp. 36–63. ISSN: 1615-5262.

[15] Peter W Shor. "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer". *SIAM J. Sci. Statist. Comput.* 26.5 (1997), pp. 1484–1509. ISSN: 0097-5397.

[16] Amos Fiat and Adi Shamir. "How to prove yourself: Practical solutions to identification and signature problems". *Conference on the Theory and Application of Cryptographic Techniques*. Springer. 1986, pp. 186–194.

[17] Richard Lindner and Chris Peikert. "Better key sizes (and attacks) for LWE-based encryption". *Cryptographers' Track at the RSA Conference*. Springer. 2011, pp. 319–339.

[18] Damien Stehlé and Ron Steinfeld. "Making NTRU as secure as worst-case problems over ideal lattices". *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer. 2011, pp. 27–47.

[19] Léo Ducas et al. "Lattice signatures and bimodal gaussians". *Advances in Cryptology–CRYPTO 2013*. Springer, 2013, pp. 40–56.

[20] Daniel J. Bernstein, Johannes Buchmann, and Erik Dahmen. *(Eds.) Post-quantum cryptography*. Berlin, Heidelberg: Springer Science and Business Media, 2009. ISBN: 3540887024.

[21] Aviad Kipnis, Jacques Patarin, and Louis Goubin. "Unbalanced oil and vinegar signature schemes". *International Conference on the Theory and Applications of Cryptographic Techniques*. Springer. 1999, pp. 206–222.

[22] Jintai Ding and Dieter Schmidt. "Rainbow, a new multivariable polynomial signature scheme". *International Conference on Applied Cryptography and Network Security*. Springer. 2005, pp. 164–175.

[23] Robert J McEliece. “A public-key cryptosystem based on algebraic”. *Coding Thv* 4244 (1978), pp. 114–116.

[24] Valerii Denisovich Goppa. “A new class of linear correcting codes”. *Problemy Peredachi Informatsii* 6.3 (1970), pp. 24–30.

[25] Ralph Merkle. “A certified digital signature”. *Advances in Cryptology—CRYPTO'89 Proceedings*. Springer. 1990, pp. 218–238.

[26] ETSI White Paper. *“Quantum Safe Cryptography and Security; An introduction, benefits, enablers and challenges"*. 2015.

[27] Johannes Buchmann, Erik Dahmen, and Andreas Hülsing. “XMSS-a practical forward secure signature scheme based on minimal security assumptions”. *International Workshop on Post-Quantum Cryptography*. Springer. 2011, pp. 117–129.

[28] Daniel J Bernstein et al. “SPHINCS: practical stateless hash-based signatures”. *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer. 2015, pp. 368–397.

[29] Howard Barnum et al. “Authentication of quantum messages”. *Proceeings of The 43rd Annual IEEE Symposium on Foundations of Computer Science, Vancouver, BC, Canada, 2002.* New York, USA: IEEE, Nov. 2002, pp. 449–458. ISBN: 0769518222.

[30] Xin Lu and Dengguo Feng. “Quantum digital signature based on quantum one-way functions”. *The 7th International Conference on Advanced Communication Technology, 2005, ICACT 2005*. Vol. 1. IEEE. 2005, pp. 514–517.

[31] Mark N Wegman and J Lawrence Carter. “New hash functions and their use in authentication and set equality”. *Journal of computer and system sciences* 22.3 (1981), pp. 265–279.

[32] Gilbert S Vernam. *Secret signaling system*. US Patent 1,310,719. 1919.

[33] Colleen M Swanson and Douglas R Stinson. “Unconditionally secure signature schemes revisited”. *Information Theoretic Security, Proceedings of ICITS 2011, LNCS, Amsterdam, The Netherlands*. Vol. 6673. Berlin, Heidelberg: Springer, 2011, pp. 100–116. ISBN: 3642207278.

[34] Juan Miguel Arrazola, Petros Wallden, and Erika Andersson. “Multiparty Quantum Signature Schemes”. *Quantum Information and Computation* 5 (2015), pp. 0435–0464.

[35] John Rompel. "One-way functions are necessary and sufficient for secure signatures". *Proceedings of the twenty-second annual ACM symposium on Theory of computing*. ACM. 1990, pp. 387–394.

[36] Leslie Lamport. *Constructing digital signatures from a one-way function*. Report. Technical Report CSL-98, SRI International Palo Alto, 1979.

[37] David Chaum and Sandra Roijakkers. "Unconditionally-secure digital signatures". *Advances in Cryptology-CRYPTO'90, LNCS, Santa Barbara, USA, 1990*. Vol. 537. Berlin, Heidelberg: Springer, 1991, pp. 206–214. ISBN: 3540545085.

[38] David Chaum. "The dining cryptographers problem: Unconditional sender and recipient untraceability". *J. Crypt.* 1.1 (1988), pp. 65–75. ISSN: 0933-2790.

[39] Birgit Pfitzmann and Michael Waidner. *Information-theoretic pseudosignatures and byzantine agreement for $t \geq n/3$*. Technical Report RZ 2882 (90830), IBM Research, 1996.

[40] Goichiro Hanaoka et al. "Unconditionally secure digital signature schemes admitting transferability". *ASIACRYPT 2000*. Springer, 2000, pp. 130–142.

[41] Junji Shikata et al. "Security notions for unconditionally secure signature schemes". *EUROCRYPT 2002*. Springer, 2002, pp. 434–449.

[42] Goichiro Hanaoka, Junji Shikata, and Yuliang Zheng. "Efficient unconditionally secure digital signatures". *IEICE transactions on fundamentals of electronics, communications and computer sciences* 87.1 (2004), pp. 120–130.

[43] Matthias Fitzi, Stefan Wolf, and Jürg Wullschleger. "Pseudo-signatures, broadcast, and multi-party computation from correlated randomness". *Annual International Cryptology Conference*. Springer. 2004, pp. 562–578.

[44] Elham Kashefi and Iordanis Kerenidis. "Statistical zero-knowledge and quantum one-way functions". *J. Theor. Comp. Sci.* 378.1 (2007), pp. 101–116. ISSN: 0304-3975.

[45] Harry Buhrman et al. "Quantum Fingerprinting". *Physical Review Letters* 87 (16 2001), p. 167902. DOI: 10.1103/PhysRevLetters87.167902.

[46] D Gavinsky and T Ito. "Quantum fingerprints that keep secrets". *J. Quant. Inf. Comp.* 13 (2013), pp. 583–606.

[47] Alexander Semenovich Holevo. "Bounds for the quantity of information transmitted by a quantum communication channel". *Prob. Inf. Trans.* 9.3 (1973), pp. 177–183. ISSN: 0555-2923.

[48] Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. Cambridge university press, 2010.

[49] Daniel Gottesman and Isaac Chuang. "Quantum digital signatures". *arXiv preprint quant-ph/0105032* (2001).

[50] Erika Andersson, Marcos Curty, and Igor Jex. "Experimentally realizable quantum comparison of coherent states and its applications". *Physical Review A* 74.2 (2006), p. 022304.

[51] Patrick Clarke et al. "Experimental demonstration of quantum digital signatures using phase-encoded coherent states of light". *Nature Communications* 3 (2012), p. 1174.

[52] Vedran Dunjko, Petros Wallden, and Erika Andersson. "Quantum digital signatures without quantum memory". *Physical Review Letters* 112.4 (2014), p. 040502.

[53] Robert Collins et al. "Realization of quantum digital signatures without the requirement of quantum memory". *Physical Review Letters* 113.4 (2014), p. 040502.

[54] Leslie Lamport, Robert Shostak, and Marshall Pease. "The Byzantine generals problem". *ACM Trans. Prog. Lang. Syst. (TOPLAS)* 4.3 (1982), pp. 382–401. ISSN: 0164-0925.

[55] Sarah Croke and Adrian Kent. "Security details for bit commitment by transmitting measurement outcomes". *Physical Review A* 86.5 (2012), p. 052309.

[56] Mark N Wegman and J Lawrence Carter. "New hash functions and their use in authentication and set equality". *J. Comp. Syst. Sci.* 22.3 (1981), pp. 265–279. ISSN: 0022-0000.

[57] Petros Wallden, Vedran Dunjko, and Erika Andersson. "Minimum-cost quantum measurements for quantum information". *Journal of Physics A: Mathematical and Theoretical* 47.12 (2014), p. 125303.

[58] Wassily Hoeffding. "Probability inequalities for sums of bounded random variables". *Journal of the American statistical association* 58.301 (1963), pp. 13–30.

[59] Ross J Donaldson et al. “Experimental demonstration of kilometer-range quantum digital signatures”. *Physical Review A* 93.1 (2016), p. 012329.

[60] Dominic Mayers. “Quantum key distribution and string oblivious transfer in noisy channels”. *Annual International Cryptology Conference*. Springer. 1996, pp. 343–357.

[61] Matthew F Pusey, Jonathan Barrett, and Terry Rudolph. “On the reality of the quantum state”. *Nature Physics* 8.6 (2012), pp. 475–478.

[62] Vašek Chvátal. “The tail of the hypergeometric distribution”. *Discrete Mathematics* 25.3 (1979), pp. 285–287.

[63] Artur K Ekert. “Quantum cryptography based on Bell’s theorem”. *Physical Review Letters* 67.6 (1991), p. 661.

[64] Renato Renner and Stefan Wolf. “Simple and tight bounds for information reconciliation and privacy amplification”. *International Conference on the Theory and Application of Cryptology and Information Security*. Springer. 2005, pp. 199–216.

[65] Marco Tomamichel, Roger Colbeck, and Renato Renner. “Duality between smooth min-and max-entropies”. *IEEE Transactions on Information Theory* 56.9 (2010), pp. 4674–4681.

[66] Carl Wilhelm Helstrom. *Quantum detection and estimation theory*. Academic press, 1976.

[67] Renato Renner. “Security of quantum key distribution”. *International Journal of Quantum Information* 6.01 (2008), pp. 1–127.

[68] Christopher A Fuchs and Jeroen Van De Graaf. “Cryptographic distinguishability measures for quantum-mechanical states”. *IEEE Transactions on Information Theory* 45.4 (1999), pp. 1216–1227.

[69] Mark M Wilde. “From classical to quantum Shannon theory”. *arXiv preprint arXiv:1106.1445* (2011).

[70] Benjamin Schumacher. “Quantum coding”. *Physical Review A* 51.4 (1995), p. 2738.

[71] Michał Horodecki, Jonathan Oppenheim, and Andreas Winter. “Quantum state merging and negative information”. *Communications in Mathematical Physics* 269.1 (2007), pp. 107–136.

[72] Renato Renner and Robert König. "Universally composable privacy amplification against quantum adversaries". *Theory of Cryptography Conference*. Springer. 2005, pp. 407–425.

[73] Marco Tomamichel and Masahito Hayashi. "A hierarchy of information quantities for finite block length analysis of quantum tasks". *IEEE Transactions on Information Theory* 59.11 (2013), pp. 7693–7710.

[74] Igor Devetak and Andreas Winter. "Classical data compression with quantum side information". *Physical Review A* 68.4 (2003), p. 042301.

[75] David Slepian and Jack Wolf. "Noiseless coding of correlated information sources". *IEEE Transactions on information Theory* 19.4 (1973), pp. 471–480.

[76] Igor Devetak and Andreas Winter. "Distillation of secret key and entanglement from quantum states". *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*. Vol. 461. 2053. The Royal Society. 2005, pp. 207–235.

[77] Imre Csiszár and Janos Korner. "Broadcast channels with confidential messages". *IEEE transactions on information theory* 24.3 (1978), pp. 339–348.

[78] Alfréd Rényi et al. "On measures of entropy and information". *Proceedings of the fourth Berkeley symposium on mathematical statistics and probability*. Vol. 1. 1961, pp. 547–561.

[79] Marco Tomamichel. "A framework for non-asymptotic quantum information theory". *arXiv preprint arXiv:1203.2142* (2012).

[80] Robert Konig, Renato Renner, and Christian Schaffner. "The operational meaning of min-and max-entropy". *IEEE Transactions on Information theory* 55.9 (2009), pp. 4337–4347.

[81] Koenraad MR Audenaert. "A sharp continuity estimate for the von Neumann entropy". *Journal of Physics A: Mathematical and Theoretical* 40.28 (2007), p. 8127.

[82] John Watrous. "Theory of Quantum Information lecture notes". *https://cs.uwaterloo.ca/watrous/~LectureNotes.html* (2008).

[83] Charles H Bennett et al. "Generalized privacy amplification". *IEEE Transactions on Information Theory* 41.6 (1995), pp. 1915–1923.

[84] Joseph M Renes and Renato Renner. "One-shot classical data compression with quantum side information and the distillation of common randomness or secret keys". *IEEE Transactions on Information Theory* 58.3 (2012), pp. 1985–1991.

[85] Normand J Beaudry and Renato Renner. "An intuitive proof of the data processing inequality". *Quantum Information & Computation* 12.5-6 (2012), pp. 432–441.

[86] Isidore I Hirschman. "A note on entropy". *American journal of mathematics* 79.1 (1957), pp. 152–156.

[87] David Deutsch. "Uncertainty in quantum measurements". *Physical Review Letters* 50.9 (1983), p. 631.

[88] Marco Tomamichel and Renato Renner. "Uncertainty relation for smooth entropies". *Physical Review Letters* 106.11 (2011), p. 110506.

[89] Marco Tomamichel et al. "Tight finite-key analysis for quantum cryptography". *Nature Communications* 3 (2012), p. 634.

[90] Stephanie Wehner and Andreas Winter. "Entropic uncertainty relations: a survey". *New Journal of Physics* 12.2 (2010), p. 025009.

[91] Gilles Brassard et al. "Security aspects of practical quantum cryptography". *International Conference on the Theory and Applications of Cryptographic Techniques*. Springer. 2000, pp. 289–299.

[92] Hoi-Kwong Lo, Xiongfeng Ma, and Kai Chen. "Decoy state quantum key distribution". *Physical Review Letters* 94.23 (2005), p. 230504.

[93] Daniel Gottesman et al. "Security of quantum key distribution with imperfect devices". *Information Theory, 2004. ISIT 2004. Proceedings. International Symposium on*. IEEE. 2004, p. 136.

[94] Jonathan Katz and Yehuda Lindell. *Introduction to modern cryptography*. CRC press, 2014.

[95] Edgar N Gilbert, F Jessie MacWilliams, and Neil JA Sloane. "Codes which detect deception". *Bell System Technical Journal* 53.3 (1974), pp. 405–424.

[96] Douglas R. Stinson. "Universal hashing and authentication codes". *Designs, Codes and Cryptography* 4.3 (1994), pp. 369–380.

[97] Bert den Boer. "A Simple and Key-Economical Unconditional Authentication Scheme." *Journal of Computer Security* 2.1 (1993), pp. 65–71.

[98] Jürgen Bierbrauer et al. "On families of hash functions via geometric codes and concatenation". *Annual International Cryptology Conference*. Springer. 1993, pp. 331–342.

[99] Hugo Krawczyk. "LFSR-based hashing and authentication". *Annual International Cryptology Conference*. Springer. 1994, pp. 129–139.

[100] Aysajan Abidin and Jan-Åke Larsson. "New universal hash functions". *Western European Workshop on Research in Cryptology*. Springer. 2011, pp. 99–108.

[101] David Chaum, Claude Crépeau, and Ivan Damgard. "Multiparty unconditionally secure protocols". *Proceedings of the twentieth annual ACM symposium on Theory of computing*. ACM. 1988, pp. 11–19.

[102] Oded Goldrcich and Ronen Vainish. "How to solve any protocol problem-an efficiency improvement". *Conference on the Theory and Application of Cryptographic Techniques*. Springer. 1987, pp. 73–86.

[103] Joe Kilian. "Founding crytpography on oblivious transfer". *Proceedings of the twentieth annual ACM symposium on Theory of computing*. ACM. 1988, pp. 20–31.

[104] Dominic Mayers. "Unconditionally secure quantum bit commitment is impossible". *Physical Review Letters* 78.17 (1997), p. 3414.

[105] Hoi-Kwong Lo. "Insecurity of quantum secure computations". *Physical Review A* 56.2 (1997), p. 1154.

[106] André Chailloux, Gus Gutoski, and Jamie Sikora. "Optimal bounds for semi-honest quantum oblivious transfer". *arXiv preprint arXiv:1310.3262* (2013).

[107] Matthias Fitzi, Nicolas Gisin, and Ueli Maurer. "Quantum solution to the Byzantine agreement problem". *Physical Review Letters* 87.21 (2001), p. 217901.

[108] Hua-Lei Yin, Yao Fu, and Zeng-Bing Chen. "Practical quantum digital signature". *Physical Review A* 93.3 (2016), p. 032316.

[109] Hua-Lei Yin et al. "Experimental Quantum Digital Signature over 102 km". *arXiv preprint arXiv:1608.01086* (2016).

[110] Ryan Amiri et al. "Secure quantum signatures using insecure quantum channels". *Physical Review A* 93.3 (2016), p. 032325.

[111] GL Roberts et al. “Experimental measurement-device-independent quantum digital signatures”. *arXiv preprint arXiv:1703.00493* (2017).

[112] Torben Pryds Pedersen. “Non-interactive and information-theoretic secure verifiable secret sharing”. *Annual International Cryptology Conference*. Springer. 1991, pp. 129–140.

[113] Tian-Yin Wang et al. “Security of quantum digital signatures for classical messages”. *Scientific reports* 5 (2015).

[114] Charles Ci Wen Lim et al. “Concise security bounds for practical decoy-state quantum key distribution”. *Physical Review A* 89.2 (2014), p. 022307.

[115] Hoi-Kwong Lo, Hoi Fung Chau, and Mohammed Ardehali. “Efficient quantum key distribution scheme and a proof of its unconditional security”. *Journal of Cryptology* 18.2 (2005), pp. 133–165.

[116] Normand J Beaudry, Tobias Moroder, and Norbert Lütkenhaus. “Squashing models for optical measurements in quantum communication”. *Physical review letters* 101.9 (2008), p. 093601.

[117] Marco Tomamichel and Anthony Leverrier. “A rigorous and complete proof of finite key security of quantum key distribution”. *arXiv preprint arXiv:1506.08458* (2015).

[118] Robert J Serfling. “Probability inequalities for the sum in sampling without replacement”. *The Annals of Statistics* (1974), pp. 39–48.

[119] Marco Tomamichel et al. “Fundamental finite key limits for information reconciliation in quantum key distribution”. *Information Theory (ISIT), 2014 IEEE International Symposium on*. IEEE. 2014, pp. 1469–1473.

[120] Valerio Scarani and Renato Renner. “Quantum cryptography with finite resources: Unconditional security bound for discrete-variable protocols with one-way postprocessing”. *Physical Review Letters* 100.20 (2008), p. 200501.

[121] Marco Lucamarini et al. “Efficient decoy-state quantum key distribution with quantified security”. *Optics express* 21.21 (2013), pp. 24550–24565.

[122] Robert J Collins et al. “Experimental transmission of quantum digital signatures over 90 km of installed optical fiber using a differential phase shift quantum key distribution system”. *Optics Letters* 41.21 (2016), pp. 4883–4886.

[123] J. Robert Collins et al. “Experimental demonstration of quantum digital signatures over 43dB channel loss using differential phase shift quantum key distribution”. *Scientific Reports* 7 (2017), p. 3235.

[124] Raymond YQ Cai and Valerio Scarani. “Finite-key analysis for practical implementations of quantum key distribution”. *New Journal of Physics* 11.4 (2009), p. 045024.

[125] Kyo Inoue, Edo Waks, and Yoshihisa Yamamoto. “Differential phase shift quantum key distribution”. *Physical Review Letters* 89.3 (2002), p. 037902.

[126] Kai Wen, Kiyoshi Tamaki, and Yoshihisa Yamamoto. “Unconditional security of single-photon differential phase shift quantum key distribution”. *Physical Review Letters* 103.17 (2009), p. 170503.

[127] Kiyoshi Tamaki, Masato Koashi, and Go Kato. “Unconditional security of coherent-state-based differential phase shift quantum key distribution protocol with block-wise phase randomization”. *arXiv preprint arXiv:1208.1995* (2012).

[128] Eleni Diamanti. “Security and implementation of differential phase shift quantum key distribution systems”. PhD thesis. Stanford University, 2006.

[129] Chi-Hang Fred Fung et al. “Phase-remapping attack in practical quantum-key-distribution systems”. *Physical Review A* 75.3 (2007), p. 032314.

[130] Feihu Xu, Bing Qi, and Hoi-Kwong Lo. “Experimental demonstration of phase-remapping attack in a practical quantum key distribution system”. *New Journal of Physics* 12.11 (2010), p. 113026.

[131] Bing Qi et al. “Time-shift attack in practical quantum cryptosystems”. *arXiv preprint quant-ph/0512080* (2005).

[132] Yi Zhao et al. “Quantum hacking: Experimental demonstration of time-shift attack against practical quantum-key-distribution systems”. *Physical Review A* 78.4 (2008), p. 042333.

[133] Lars Lydersen et al. “Hacking commercial quantum cryptography systems by tailored bright illumination”. *Nature photonics* 4.10 (2010), pp. 686–689.

[134] I Gerhardt et al. “Full-field implementation of a perfect eavesdropper on a quantum cryptography system.” *Nature Communications* 2 (2010), pp. 349–349.

[135] Nicolas Gisin, Stefano Pironio, and Nicolas Sangouard. "Proposal for implementing device-independent quantum key distribution based on a heralded qubit amplifier". *Physical Review Letters* 105.7 (2010), p. 070501.

[136] Marcos Curty and Tobias Moroder. "Heralded-qubit amplifiers for practical device-independent quantum key distribution". *Physical Review A* 84.1 (2011), p. 010304.

[137] Ittoop Vergheese Puthoor et al. "Measurement-device-independent quantum digital signatures". *Physical Review A* 94.2 (2016), p. 022328.

[138] Hoi-Kwong Lo, Marcos Curty, and Bing Qi. "Measurement-device-independent quantum key distribution". *Physical Review Letters* 108.13 (2012), p. 130503.

[139] Eli Biham, Bruno Huttner, and Tal Mor. "Quantum cryptographic network based on quantum memories". *Physical Review A* 54.4 (1996), p. 2651.

[140] Marcos Curty et al. "Finite-key analysis for measurement-device-independent quantum key distribution". *Nature Communications* 5 (2014).

[141] Masahiro Takeoka, Saikat Guha, and Mark M Wilde. "Fundamental rate-loss tradeoff for optical quantum key distribution". *Nature Communications* 5 (2014).

[142] LC Comandar et al. "Quantum key distribution without detector vulnerabilities using optically seeded lasers". *Nature Photonics* (2016).

[143] F Marsili et al. "Detecting single infrared photons with 93% system efficiency". *Nature Photonics* 7.3 (2013), pp. 210–214.

[144] Rupert Ursin et al. "Entanglement-based quantum communication over 144 km". *Nature physics* 3.7 (2007), pp. 481–486.

[145] LC Comandar et al. "Quantum key distribution without detector vulnerabilities using optically seeded lasers". *Nature Photonics* (2016).

[146] Lucian C Comandar et al. "Gigahertz-gated InGaAs/InP single-photon detector with detection efficiency exceeding 55% at 1550 nm". *Journal of Applied Physics* 117.8 (2015), p. 083109.

[147] Adi Shamir. "How to share a secret". *Communications of the ACM* 22.11 (1979), pp. 612–613.

[148] Ryan Amiri et al. "Unconditionally secure signatures". *Cryptology ePrint Archive,(Report 2016/739)* (2016).

[149] Mustafa Atici and D Stinson. "Universal hashing and multiple authentication". *Advances in Cryptology—CRYPTO'96*. Springer. 1996, pp. 16–30.

[150] Aysajan Abidin and Jan-Åke Larsson. "New universal hash functions". *Western European Workshop on Research in Cryptology*. Springer. 2011, pp. 99–108.

[151] M Lucamarini et al. "Efficient decoy-state quantum key distribution with quantified security". *Optics express* 21.21 (2013), pp. 24550–24565.

[152] Yishay Mansour, Noam Nisan, and Prasoon Tiwari. "The computational complexity of universal hashing". *Theoretical Computer Science* 107.1 (1993), pp. 121–133.

[153] Ryan Amiri, Petros Wallden, and Erika Andersson. "Almost tight lower bounds for 1-out-of-2 quantum oblivious transfer". *Manuscript submitted* ().

[154] Stephen Wiesner. "Conjugate coding". *ACM Sigact News* 15.1 (1983), pp. 78–88.

[155] Shimon Even, Oded Goldreich, and Abraham Lempel. "A randomized protocol for signing contracts". *Communications of the ACM* 28.6 (1985), pp. 637–647.

[156] Michael O Rabin. "How To Exchange Secrets with Oblivious Transfer." *IACR Cryptology ePrint Archive* 2005 (2005), p. 187.

[157] Claude Crépeau. "Equivalence between two flavours of oblivious transfers". *Conference on the Theory and Application of Cryptographic Techniques*. Springer. 1987, pp. 350–354.

[158] Gilles Brassard and Claude Crépeau. "Oblivious transfers and privacy amplification". *International Conference on the Theory and Applications of Cryptographic Techniques*. Springer. 1997, pp. 334–347.

[159] Gilles Brassard, Claude Crépeau, and Stefan Wolf. "Oblivious transfers and privacy amplification". *Journal of Cryptology* 16.4 (2003), pp. 219–237.

[160] Ivan B Damgård et al. "Cryptography in the bounded-quantum-storage model". *SIAM Journal on Computing* 37.6 (2008), pp. 1865–1890.

[161] Alexei Kitaev. "Quantum coin-flipping". *Talk at QIP* (2003).

[162] Carlos Mochon. "Quantum weak coin flipping with arbitrarily small bias". *arXiv preprint arXiv:0711.4114* (2007).

[163] André Chailloux and Iordanis Kerenidis. "Optimal quantum strong coin flipping". *Foundations of Computer Science, 2009. FOCS'09. 50th Annual IEEE Symposium on*. IEEE. 2009, pp. 527–533.

[164] André Chailloux and Iordanis Kerenidis. "Optimal bounds for quantum bit commitment". *Foundations of Computer Science (FOCS), 2011 IEEE 52nd Annual Symposium on*. IEEE. 2011, pp. 354–362.

[165] André Chailloux, Iordanis Kerenidis, and Jamie Sikora. "Lower bounds for quantum oblivious transfer". *arXiv preprint arXiv:1007.1875* (2010).

[166] André Chailloux, Iordanis Kerenidis, and Jamie Sikora. "Lower bounds for quantum oblivious transfer". *arXiv preprint arXiv:1007.1875* (2010).

[167] Louis Salvail, Christian Schaffner, and Miroslava Sotáková. "On the power of two-party quantum cryptography". *International Conference on the Theory and Application of Cryptology and Information Security*. Springer. 2009, pp. 70–87.

[168] Paul Hausladen and William K Wootters. "A pretty good measurement for distinguishing quantum states". *Journal of Modern Optics* 41.12 (1994), pp. 2385–2390.

[169] Koenraad MR Audenaert and Milán Mosonyi. "Upper bounds on the error probabilities and asymptotic error exponents in quantum multiple state discrimination". *Journal of Mathematical Physics* 55.10 (2014), p. 102201.

[170] Armin Uhlmann. "The "transition probability" in the state space of a*-algebra". *Reports on Mathematical Physics* 9.2 (1976), pp. 273–279.

[171] Scott Aaronson. "Quantum copy-protection and quantum money". *Computational Complexity, 2009. CCC'09. 24th Annual IEEE Conference on*. IEEE. 2009, pp. 229–242.

[172] Andrew Lutomirski. "An online attack against Wiesner's quantum money". *arXiv preprint arXiv:1010.0256* (2010).

[173] Aharon Brodutch et al. "An adaptive attack on Wiesner's quantum money". *Quantum Information and Computation* 16 (11&12 2016), pp. 1048–1070.

[174] Dmitry Gavinsky. "Quantum money with classical verification". *Computational Complexity (CCC), 2012 IEEE 27th Annual Conference on*. IEEE. 2012, pp. 42–52.

[175] Ziv Bar-Yossef, Thathachar S Jayram, and Iordanis Kerenidis. "Exponential separation of quantum and classical one-way communication complexity". *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*. ACM. 2004, pp. 128–137.

[176] Fernando Pastawski et al. "Unforgeable noise-tolerant quantum tokens". *Proceedings of the National Academy of Sciences* 109.40 (2012), pp. 16079–16082.

[177] Marios Georgiou and Iordanis Kerenidis. "New Constructions for Quantum Money". *LIPIcs-Leibniz International Proceedings in Informatics*. Vol. 44. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. 2015.

[178] Edward Farhi et al. "Quantum state restoration and single-copy tomography for ground states of hamiltonians". *Physical Review Letters* 105.19 (2010), p. 190503.

[179] Edward Farhi et al. "Quantum money from knots". *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*. ACM. 2012, pp. 276–289.

[180] Andrew Lutomirski et al. "Breaking and making quantum money: toward a new quantum cryptographic protocol". *arXiv preprint arXiv:0912.3825* (2009).

[181] Ryan Amiri and Juan Miguel Arrazola. "Quantum money with nearly optimal error tolerance". *Physical Review A* 95.6 (2017), p. 062334.

[182] Scott Aaronson and Paul Christiano. "Quantum money from hidden subspaces". *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*. ACM. 2012, pp. 41–60.

[183] Juan Miguel Arrazola, Markos Karasamanis, and Norbert Lütkenhaus. "Practical quantum retrieval games". *Physical Review A* 93.6 (2016), p. 062311.

[184] Michael Keyl and Reinhard F Werner. "Optimal cloning of pure states, testing single clones". *Journal of Mathematical Physics* 40.7 (1999), pp. 3283–3299.

[185] Juan Miguel Arrazola and Norbert Lütkenhaus. "Quantum communication with coherent states and linear optics". *Physical Review A* 90.4 (2014), p. 042335.

[186] Gustavo Rigolin. "Quantum teleportation of an arbitrary two-qubit state and its relation to multipartite entanglement". *Physical Review A* 71.3 (2005), p. 032303.