
STUDIES ON THE QUANTUM PRIVATE QUERY PRIMITIVE IN THE DEVICE-INDEPENDENT PARADIGM

A thesis submitted to Indian Statistical Institute
in partial fulfillment of the thesis requirements for the degree of
Doctor of Philosophy in Computer Science

Author:

Jyotirmoy BASAK
jyotirmoy_r@isical.ac.in

Supervisor:

Prof. Subhamoy MAITRA
subho@isical.ac.in

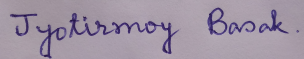


Applied Statistics Unit
Indian Statistical Institute
203, B. T. Road, Kolkata,
West Bengal, India - 700 108.

To my Parents and Grandmother

DECLARATION OF AUTHORSHIP

I, *Jyotirmoy Basak*, a research scholar at *Applied Statistics Unit (ASU), Indian Statistical Institute (ISI), Kolkata*, declare that this thesis encompasses the research conducted by me under the guidance of *Prof. Subhamoy Maitra (ASU, ISI Kolkata)*. I affirm that this work is entirely original, both in terms of research content and narrative, and to the best of my knowledge, the materials contained in this thesis have not previously been published or written by any other person, nor has been submitted as a whole or as a part for any degree/diploma or any other academic award anywhere before.

A rectangular box containing a handwritten signature in purple ink that reads "Jyotirmoy Basak."

Jyotirmoy Basak

Applied Statistics Unit
Indian Statistical Institute, Kolkata
203, Barrackpore Trunk Road
Kolkata 700108, INDIA.

ACKNOWLEDGEMENT

Every Ph.D. is perhaps an emotional journey, and mine was no exception. Here, I would like to express my sincere gratitude to all the individuals without whose guidance, inspiration, and assistance this thesis would not have been possible.

First and foremost, I am deeply indebted to my mentor and supervisor, Prof. Subhamoy Maitra, for his invaluable guidance and support throughout my academic journey at Indian Statistical Institute (ISI). It will always remain a mystery to me what exactly influenced him to make the decision to put his faith in me at the lowest point of my professional career and to take the responsibility of overseeing my dissertation, but I am immensely grateful to him for that decision, as without him, I would be nowhere.

I am profoundly grateful to Dr. Kaushik Chakraborty (Kaushik da) and Dr. Arpita Maitra (Arpita di) for their exceptional support and guidance throughout my Ph.D. journey. They have been like co-supervisors to me, offering invaluable insights, motivation, and prompt responses to my inquiries, even during unconventional hours. Their unwavering assistance has played a vital role in shaping my research.

I express my sincere appreciation to the esteemed faculty members at ISI, Kolkata, for their unwavering support throughout my tenure at the institute. I consider myself incredibly fortunate to have been mentored by such exceptional individuals during my M.Tech and Ph.D at ISI Kolkata. Specifically, I wish to offer my gratitude to Prof. Guruprasad Kar (Guru Sir) for introducing me to the fascinating subject of “Quantum information and quantum computation” during my M. Tech curriculum, igniting my interest in this field.

I am deeply grateful to all the non-teaching staff at the Applied Statistics Unit (ASU) office (specifically Arijit da, Somenath da, Amarjit da, Subrata da, and Nayan da) and the Dean’s office for their invaluable assistance.

Throughout my time at ISI, I have been fortunate to have amazing friends and seniors. I cherish the extensive moments spent with Avijit da, Nilanjan da, Aniruddha da, Subhadip da, Susanta, Laltu da, Amit da, Diptendu da, Ashwin da, Animesh da, Suman, Gourab and Chandranan. I am glad to have seniors like Avik da, Binanda da, Butu da, Srimanta da, and Sanjay da. I also enjoyed the company of colleagues and juniors like Arghya da, Anik da, Rakesh, Debendra, Subhra, Subha and Bikshan. I apologize to those I missed here, but contributed to my wonderful experience at ISI.

Life is not complete without friends, particularly the eccentric ones. I am lucky to have Satanik, Aurodip, Shamim, Debayan, Akku, and Taniya as my constant support team. I am deeply grateful to them for our numerous gatherings, endless conversations on diverse subjects, and their constant motivation during moments of discouragement. Their enduring friendship and support alleviated my stress on multiple occasions.

Lastly, my heartfelt gratitude goes to my parents and grandmother for their unwavering support and understanding and the numerous sacrifices they have made for me. No words are sufficient to repay their gratitude. I am extremely thankful to them for trusting me and being there throughout the highs and lows. I am also thankful to the other family members for their continuous support, love, and help.

ABSTRACT

“It’s still magic even if you know how it’s done.”

— Terry Pratchett, *A Hat Full of Sky*.

In this thesis, we focus on the Quantum Private Query (QPQ) primitive in the device-independent (DI) paradigm, addressing the challenges of preserving user and database privacy without trusting the devices. Existing cryptographic primitives, such as Symmetric Private Information Retrieval (SPIR) and 1 out of N Oblivious Transfer (OT), lack unconditional security with a single server in both classical and quantum domains. The QPQ primitive addresses this limitation by allowing the client to gain probabilistic knowledge about unintended data bits while expecting the server not to cheat if a non-zero probability exists of being caught.

The contributions of this thesis include proposing and analyzing QPQ schemes within the DI framework. We introduce a novel QPQ scheme using EPR pairs, exploiting self-testing of shared Bell states, projective measurement operators, and a specific class of POVM operators to achieve complete device independence. We address the limitations of a semi-DI-QPQ proposal and utilize the tilted version of the actual CHSH game and self-testing of observables to enhance security and certify full device independence. Furthermore, we suggest several strategies to reduce the overall sample size required for DI testing of that semi-DI-QPQ proposal in the finite sample scenario. Moreover, we address the limitations of the existing multi-user QPQ schemes and propose a semi-DI multi-user QPQ scheme where each user can retrieve different items simultaneously without revealing their choices to others or relying on a semi-trusted server. We formally conduct security assessments for all our DI-QPQ proposals and derive upper limits on the cheating probabilities to ensure robust DI-QPQ implementations.

Overall, in this thesis, we contribute to advancing the QPQ primitive in the DI paradigm, offering novel schemes and addressing the challenges posed by distrustful settings and multi-user scenarios.

LIST OF PUBLICATIONS/MANUSCRIPTS

Following are the list of publications/manuscripts that are included in this thesis. Chapter 4 is based on the papers 1 & 2. Chapter 5 is based on the paper 3. Chapter 6 is based on the paper 4 and Chapter 7 is based on the paper 5.

1. **Jyotirmoy Basak**, Kaushik Chakraborty, Arpita Maitra and Subhamoy Maitra.
“A Proposal for Device Independent Probabilistic Quantum Oblivious Transfer”.
Progress in Cryptology - INDOCRYPT 2022, LNCS, vol 13774, pp 541–565,
Springer, Cham.
DOI: https://doi.org/10.1007/978-3-031-22912-1_24
2. **Jyotirmoy Basak**, Kaushik Chakraborty, Arpita Maitra and Subhamoy Maitra.
“Improved and Formal Proposal for Device Independent Quantum Private Query”.
Preprint: <https://arxiv.org/abs/1901.03042>
(This is the full version of paper 1. This work has been presented as a talk in
AQIS 2020 conference and as a poster in QCrypt 2021 conference).
3. **Jyotirmoy Basak** and Kaushik Chakraborty.
“Fully Device Independent Quantum Private Query”.
(This work has been presented as a poster in QIP 2021 and TQC 2021 confer-
ence).
4. **Jyotirmoy Basak** and Subhamoy Maitra.
“Clauser-Horne-Shimony-Holt versus three-party pseudo-telepathy: on the opti-
mal number of samples in device-independent quantum private query.”
Quantum Information Processing. 17, 77 (2018).
DOI: <https://doi.org/10.1007/s11128-018-1849-2>
5. **Jyotirmoy Basak**.
“Multi-User Semi Device Independent Quantum Private Query”.
Quantum Information Processing. 22, 276 (2023).
DOI: <https://doi.org/10.1007/s11128-023-04028-8>

Contents

Abstract	i
List of Publications/Manuscripts	ii
List of Figures	vi
List of Tables	viii
List of Acronyms and Abbreviations	x
List of Symbols	xi
1 Introduction	1
1.1 Information-theoretic and computational security	2
1.2 Two party cryptography	3
1.3 Assumptions for two-party cryptography	5
1.4 Quantum cryptography	6
1.5 Device independent quantum cryptography	9
1.6 Contribution and organization of the thesis	10
2 Preliminaries and Background	13
2.1 Basics of quantum computation	13
2.1.1 Quantum bits or qubits	13
2.1.2 Operations on qubits	15
2.1.3 Mixed states	19
2.1.4 Entanglement	20
2.1.5 Distance measures between quantum states	21
2.1.6 Distinguishability of quantum states	22
3 Quantum Private Query	25
3.1 Overview	25
3.2 Relation between QPQ, SPIR and OT	26
3.3 Comparison with the exact classical primitive	28
3.4 Evolution of Quantum Private Query	29
3.5 Security definitions	31

3.6	Security assumptions	33
4	Improved and Formal Proposal for Fully Device Independent QPQ using EPR Pairs	35
4.1	Contribution of this chapter	36
4.2	Our DI-QPQ proposal	37
4.3	Analysis of the protocol	45
4.3.1	Correctness of the protocol	45
4.3.2	Estimation of parameters for private query phase	47
4.3.3	Security of the protocol	52
4.4	Choice of initial sample size in practice	65
4.5	Statement and proof of Theorem 2	67
4.6	Verification of Alice’s POVM elements	74
4.7	Correctness of the scheme considering devices “up to a unitary”	83
4.8	Discussion and Conclusion	85
5	Proposal For Fully Device Independent QPQ using Non-maximally Entangled States	87
5.1	Revisiting the QPQ scheme [117] and its DI version in [77]	88
5.2	Contribution of this chapter	89
5.3	An attack on the DI-QPQ scheme [77]	90
5.4	Full DI proposal for the QPQ scheme [117]	93
5.4.1	Proposed full DI version of the scheme [117]:	93
5.4.2	Analysis of our scheme	98
5.4.3	Comparison with the QPQ scheme mentioned in Chapter 4	111
5.5	Full DI proposal for a modified version of the QPQ scheme [117]	113
5.5.1	Modified full DI protocol	114
5.5.2	Analysis of the modified scheme	117
5.6	Statement and proof of Theorem 8	124
5.7	Statement and proof of Theorem 9	127
5.8	Statement and proof of Theorem 13	134
6	Finite Sample Analysis in Device Independent QPQ	139
6.1	Background	140
6.1.1	CHSH and Parity Game	140
6.1.2	Estimation of Sample Size For Finite Sample Scenario	141
6.1.3	Device Independence in QPQ	141
6.2	Contribution of this chapter	144
6.3	Analysis of CHSH game with modified two-qubit entangled states	144
6.3.1	Success probability calculation	144
6.3.2	Appropriate choice of measurement basis	145

6.4	Analysis of three-party quantum pseudo telepathy with transformed three-qubit entangled states	146
6.4.1	Transformation of two-qubit state into three-qubit	147
6.4.2	Comparative study	148
6.4.3	Towards security analysis for finite samples	150
6.5	Discussion and Conclusion	151
7	Proposal For Multi-User Semi Device Independent QPQ	153
7.1	Limitations of the existing multi-user QPQ proposals	154
7.2	Contribution of this chapter	154
7.3	Our semi-DI-QPQ proposal	155
7.4	Analysis of our proposal	161
7.4.1	Correctness of the scheme	161
7.4.2	Estimation of parameters for private query phase	164
7.4.3	Security issues of the scheme	167
7.5	Discussion and Conclusion	170
8	Conclusion	171
8.1	Summary of technical results	171
8.2	Possible future works	172

List of Figures

1-1	Line diagram representing the contribution of this thesis towards the evolution in the field of QPQ	11
2-1	Representation of a qubit on a unit circle	15
2-2	Pure states and the corresponding measurement basis for Neumark's measurement	23
3-1	Line Diagram for the relations between QPQ, PIR, and SPIR	27
4-1	Visual representation of different steps of our DI-QPQ proposal.	43
4-2	Comparison between maximum inconclusive and conclusive success probability of the client.	63
5-1	Evolution of QPQ schemes in DI scenario	88
5-2	Comparison between the success probabilities of getting a raw key bit using projective and POVM measurements	92
5-3	Schematic diagram of the semi DI-QPQ scheme [77] (left) and our proposed full DI version of [117] (right)	99
5-4	Comparison between the fraction of samples used for raw key generation in two different protocols for different values of γ	113
5-5	Visual representation of our modified DI-QPQ scheme	117
6-1	Circuit diagram for transformed state	148
6-2	Comparative study of success probabilities between CHSH and parity game for DI-QPQ protocol	149
7-1	Schematic diagram of different phases of our multi-client Semi-DI QPQ scheme	161

List of Tables

2.1	Basic notations used in quantum computation	14
3.1	Known results for (S)PIR protocols in single server scenario	28
5.1	Probabilities of Different POVM Outcomes	91

LIST OF ACROYNMS AND ABBREVIATIONS

Expansion	Acronyms/ Abbreviations
Quantum Private Query	QPQ
Device Independent	DI
Oblivious Transfer	OT
Private Information Retrieval	PIR
Symmetric Private Information Retrieval	SPIR
Quantum Key Distribution	QKD
Independent and Identically Distributed	<i>i.i.d.</i>
Einstein-Podolsky-Rosen	EPR
Clauser-Horne-Shimony-Holt	CHSH
Exclusive-OR	XOR
That is	<i>i.e.</i>

LIST OF SYMBOLS

Here we describe all the notations that are used throughout this thesis.

- \mathcal{K} : The total number of initial states for our every proposal. It is assumed to be asymptotically large.
- \mathbb{I}_k : k dimensional identity matrix.
- $\mathcal{A}(\mathcal{A}^*)$: The client Alice with honest (dishonest) behavior in single-user single-server QPQ proposal.
- $\mathcal{B}(\mathcal{B}^*)$: the server Bob with honest (dishonest) behavior in single-user single-server QPQ proposal.
- $\mathcal{A}_i(\mathcal{A}_i^*)$: Subsystem of honest (dishonest) client Alice corresponding to the i -th shared state in single-user single-server QPQ proposal.
- $\mathcal{B}_i(\mathcal{B}_i^*)$: Subsystem of honest (dishonest) server Bob corresponding to the i -th shared state in single-user single-server QPQ proposal.
- $\mathcal{C}_i(\mathcal{C}_i^*)$: Honest (dishonest) i -th client in multi-user single-server QPQ proposal.
- $\mathcal{S}(\mathcal{S}^*)$: Honest (dishonest) server in multi-user single-server QPQ proposal.
- $|\phi\rangle_{\mathcal{A}_i\mathcal{B}_i}$: The i th shared state in single-user single-server QPQ proposal with the first qubit belonging to the client Alice (\mathcal{A}_i) and the second to the server Bob (\mathcal{B}_i).
- $\rho_{\mathcal{A}_i\mathcal{B}_i}$: The density matrix of the i th state in single-user single-server QPQ proposal.
- $\rho_{\mathcal{A}_i}(\rho_{\mathcal{B}_i})$: The reduced density matrices for the client Alice and the server Bob, respectively, of the i th state in single-user single-server QPQ proposal.
- X : The database held by the server, with N bits.
- $R(R_{\mathcal{A}})$: The entire raw key at Bob's and Alice's sides, respectively, each with kN bits.
- $F(F_{\mathcal{A}})$: The entire final key at Bob's and Alice's sides, respectively, each with N bits.
- $R_{\mathcal{C}}^i$: The entire raw key at the i -th client's side in multi-user single-server QPQ proposal, with kN bits.
- $F_{\mathcal{C}}^i$: The entire final key at the i -th client's side in multi-user single-server QPQ proposal, with N bits.
- R_i : The i th raw key bit of the server.

- F_i : The i th final key bit of the server.
- $R_{\mathcal{A}_i}$: The i th raw key bit of the client Alice in single-user single-server QPQ proposal.
- $F_{\mathcal{A}_i}$: The i th final key bit of the client Alice in single-user single-server QPQ proposal.
- $R_{\mathcal{C}_j}^i$: The j -th raw key bit at the i -th client's side in multi-user single-server QPQ proposal.
- $F_{\mathcal{C}_j}^i$: The j -th final key bit at the i -th client's side in multi-user single-server QPQ proposal.
- r_i : The i -th random bit chosen at the server's side in multi-user single-server QPQ proposal, which serves as the i -th raw key bit i.e., $R_i = r_i$.
- k : Number of raw key bits combined by XOR to produce each bit of the final key.
- \mathcal{I}_l : The set of query indices made by the client Alice in single-user single-server QPQ proposal.
- l : Size of the query set \mathcal{I}_l of the client Alice in single-user single-server QPQ proposal.
- \mathcal{I}_i : The set of query indices of the i -th client in multi-user single-server QPQ proposal.
- l_i : Size of the query set \mathcal{I}_i of the i -th client in multi-user single-server QPQ proposal.
- a_i : the classical bit announced by the server for i -th shared state.
- $A(B)$: The client Alice and the server Bob's measurement outcomes, respectively in single-user single-server QPQ proposal.
- $S(C_i)$: The measurement outcome at the server's (i -th client's) side in multi-user single-server QPQ proposal.
- $|0'\rangle = \cos \theta |0\rangle + \sin \theta |1\rangle$.
- $|1'\rangle = \sin \theta |0\rangle - \cos \theta |1\rangle$.
- \in_R : uniform random selection from a set.

Introduction

“Quantum cryptography would mark the end of the battle between codemakers and codebreakers, the codemakers emerging victorious, because quantum cryptography is a truly unbreakable system of encryption.”

— Simon Singh, *The Code Book: The Secrets Behind Codebreaking*.

Cryptography involves techniques for secure communication over an insecure channel in the presence of third parties called *adversaries*. Historically, the term cryptography develops from two Greek words, *kryptós* meaning “secret” and *graphein* meaning “to write”. Throughout human history, people have recognized the importance of keeping information secret, especially in the contexts of military, diplomatic, and other sensitive communication. Evidence of encryption dates back some 4000 years to hieroglyphic inscriptions in ancient Egypt. The two world wars, the cold war, and the rise of the Internet have all spurred rapid advancements in cryptography. A comprehensive history and analysis of cryptography can be found in [64]. As technology continues to evolve, the field of cryptography continues to expand beyond privacy and confidentiality, now encompassing data integrity and authentication to meet security needs in various public domains.

From a designer’s perspective, a cryptographic scheme should ideally maintain its intended functionality despite repeated attempts by malicious third parties to compromise it. This requires the proposal to be secure without relying on any assumptions about the operational environment, and not merely resistant to specific types of attacks. Many cryptographic schemes that claim to be secure under certain assumptions can be shown to be vulnerable to more general attacks. For example, the *Caesar cipher* can be easily broken using frequency analysis. This highlights the importance of defining security notions based on rigorous mathematical foundations.

Before 1948-49, most cryptographic schemes were developed based on heuristic and ad hoc approaches [6, 5, 3]. In 1948-49, Claude Shannon’s seminal work in his two landmark papers from that period first introduced the notion of mathematical cryptography. In his first paper [99], he laid the foundations of information theory,

while in the second paper [100] titled “Communication Theory of Secrecy Systems,” he presented the first concrete mathematical treatment of the field of cryptography. This work focused on two primary objectives: *secrecy* and *authenticity*. *Secrecy* in cryptography ensures that only authorized users can access a message, while *authenticity* ensures that the message can only be created and sent by a legitimate source and cannot be tampered with by anyone else. In 1976, Diffie and Hellman in their paper “New Directions in Cryptography” [45], identified the requirement of data integrity, authentication, and non-repudiation in cryptographic protocols. Modern cryptography has incorporated all these elements along with the traditional need for confidentiality. In fact, this work by Diffie and Hellman in 1976, is the first formal exposition of Public Key Cryptography. However, the real breakthrough in cryptography came when Rivest, Shamir, and Adleman discovered an amazingly simple scheme for encryption, popularly known as RSA encryption [94]. From security aspects, modern cryptographic schemes are mainly divided into two types, namely *information-theoretic secure schemes* and *computational secure schemes*.

1.1 Information-theoretic and computational security

Claude Shannon’s seminal work in his two landmark papers in 1948-49 introduced the notion of *information-theoretic security* in cryptography.

A cryptosystem is said to be ***perfectly secure*** or ***unconditionally secure*** or ***information theoretically secure*** if an adversary, even with unlimited computational power, can’t gain any information about the secret.

That is why, such a system is also called cryptanalytically unbreakable. The one-time pad is an example of an information-theoretically secure cryptographic scheme, where two parties share a secret key that has the same length as the message they want to send securely. The encryption and decryption of this message simply involve XORing the key to the message. Although the one-time pad offers perfect security, the implementation of this scheme is challenging in the sense that it requires a long secret key that can’t be reused. For this reason, in most cases, information-theoretically secure cryptosystems are only used for the most sensitive governmental communications, such as diplomatic cables and high-level military communications. In practice, it is desirable for most daily life applications to remain secure for a certain period (instead of security for indefinite times) against adversaries that do not possess unlimited capabilities. That is why, another security notion was introduced later called *computational security*.

A cryptosystem is considered ***computationally secure*** if an attacker with limited computational resources is unable to gain any information about the secret.

The achievement of computational security typically involves reducing the problem of breaking the cryptosystem to solving a problem that is believed to be computationally difficult. The security of the cryptosystem is then derived based on the fact that no polynomial time algorithm (polynomial relative to the size of the input parameter) exists yet to solve the problem with limited computational resources. In general, proving that a problem can not be solved in polynomial time is often difficult. So, one relies on computational assumptions to establish the security of a cryptosystem. For example, one can consider the security of RSA cryptosystem [94]. Although it is not known whether the RSA cryptosystem was designed considering the hardness assumption of the prime factorization problem, recently it was shown in [32] that “breaking RSA may be as difficult as factoring”. However, for the cryptosystems designed based on some hard problems, the designers should be careful in choosing the underlying problems as some cryptosystems have been broken recently because of the vulnerability of the underlying problems. An example is the Merkle-Hellman public key cryptosystem [83], which was based on the knapsack problem and has been cryptanalyzed recently by several attacks [98, 10]. Moreover, advancements in technology and computing models can render certain cryptographic systems obsolete. For example, the RSA cryptosystem is still considered secure against classical attacks, but it becomes vulnerable in the quantum paradigm because of Shor’s algorithm [102]. Similarly, the vulnerability of the discrete log problem in the quantum paradigm also causes the Diffie-Hellman key exchange scheme to become insecure in the quantum scenario.

All these examples illustrate that along with the information-theoretic secure schemes which are (in general) hard to design, even the construction of conditional secure schemes become challenging with the invention of quantum computers. This has led researchers to focus on developing quantum-secure cryptographic schemes, which can resist attacks even in the presence of quantum computers. To ensure security in the age of quantum computing, researchers have developed quantum secure schemes both in the classical as well as in the quantum domain. Quantum cryptography (such as quantum key distribution, quantum oblivious transfer, quantum bit commitment, etc.) involves designing secure schemes in the quantum domain by leveraging the unique properties of quantum mechanics, while post-quantum cryptography (such as lattice-based cryptography, code-based cryptography, multivariate cryptography, etc.) involves designing secure schemes in the classical domain by exploiting some hard mathematical problems that are believed to be difficult to solve even for quantum computers. This thesis is focused on *Quantum Cryptography*, which is both a challenging and fascinating interdisciplinary field of research in the current scenario.

1.2 Two party cryptography

The field of cryptography has evolved to provide solutions for various applications. However, this thesis is solely focused on two-party cryptography, which deals with scenarios where two parties, Alice and Bob, are involved in a task but do not fully trust each other. In such scenarios, it is crucial to minimize the amount of information

revealed during the protocol. An example of such a scenario is the millionaires' problem, introduced by Andrew Yao [118], in which two millionaires need to determine who is richer without disclosing their actual wealth. This problem is relevant in everyday life, and there are other similar scenarios, some of which are listed below.

- **Oblivious Transfer** : Suppose a user Alice wants to download a movie from an online movie service, and Bob is the server handling the service. The service charges per downloaded movie, and Alice has paid for one movie but is concerned about privacy and does not want to reveal her choice to Bob. Bob, however, wants to ensure that Alice only downloads the movie she has paid for and does not access the entire movie database. This problem is known as oblivious transfer, which is an important building block for two-party cryptography. It can be used to construct any other two-party primitive [70].
- **Bit Commitment** : Suppose Alice wants to bid in an auction hosted by Bob, but she doesn't want to reveal her bid until the auction is open. To address this issue, a commitment scheme is used, which is a type of primitive in two-party cryptography. The simplest form of a commitment scheme [26, 29], known as a bit commitment, allows Alice to commit to a single bit without revealing its value to anyone, including Bob. A secure commitment scheme must satisfy two properties: it must be hiding, meaning Bob cannot learn any information about the bid before the opening phase, and binding, meaning Alice cannot change the value of her bid after the commitment. This ensures that the auction is fair and that neither party gains an unfair advantage.
- **Position-based cryptography**: In this scenario introduced in [36], the sole credential used to access certain information is the geographic location of a party. This type of authentication could be useful in a military context where it is necessary to ensure that orders from the headquarters can only be accessed by someone physically present inside the army base at a specific location, rather than by enemies in the surrounding area.

In all the tasks mentioned above, the two parties have conflicting interests, which makes it challenging to ensure security for both parties simultaneously. Ensuring complete protection for one party would inevitably leave the other party completely unprotected. For instance, in the case of oblivious transfer, Alice can sacrifice her privacy and reveal which movie she wants to watch, or Bob can provide Alice with the entire database, hoping that she won't misuse his trust. However, these solutions come at a cost and fail to satisfy both parties' requirements. Therefore, the challenge is to devise protocols that offer a reasonable trade-off between security and functionality to ensure the security of both parties.

Designing information-theoretically secure cryptographic schemes for two-party cryptography is generally considered impossible [41]. One way to overcome this challenge is to change the security model. For instance, in the case of a key exchange where two parties want to share a secret over an insecure channel, it is impossible to use only classical information. However, it becomes possible if Alice and Bob use

quantum information, as demonstrated by the protocol developed by Charlie Bennett and Gilles Brassard in their seminal paper on quantum key distribution (QKD) [21]. This discovery paved the way for a new research direction, known as quantum cryptography.

In addition to quantum mechanics, other physical theories, such as the special theory of relativity, can be utilized to achieve tasks with information-theoretic security. While certain two-party cryptographic primitives, such as bit commitment, cannot be achieved solely through quantum mechanics, they can be obtained by leveraging the restrictions imposed by these physical theories. Such theories can limit the power and resources of an adversary. Similarly, it is possible to define new security models, such as the bounded storage and noisy storage models, by considering the current limitations of technology, for instance, the unavailability of ideal quantum memories.

1.3 Assumptions for two-party cryptography

Regrettably, it has been proven that achieving information-theoretic security for both parties in two-party cryptographic protocols is not possible (both in classical as well as in quantum domain) without additional assumptions [41, 74]. Here we provide a concise summary of several reasonable assumptions that enable the realization of information-theoretically secure two-party cryptography.

- **Assumption about trusted third-party :** Introducing a trusted third party is a simple solution for implementing any two-party primitive for Alice and Bob. However, it is not satisfactory in a scenario where Alice and Bob do not trust themselves. This solution also makes all tasks trivially possible.
- **Assumption on pre-shared resources :** For two-party cryptography, a security guarantee can be defined under the assumption that the adversaries have limited resources. Typically, we make assumptions about the following resources:

- **Memory :**

Bounded storage model : This model puts a restriction on the amount of quantum or classical memory that an adversary can use. In the case of quantum memory, this model was first introduced by Damgard et al. [44] and it forces adversaries to convert some of their quantum information into classical information, which may lead to the irreversible destruction of some of their information. This model can be used to design information-theoretically secure bit commitment and oblivious transfer protocols, as shown in [44, 112].

Noisy storage model : In this scenario, we assume that the quantum memory used by the adversary to store quantum states (or the information) is affected by noise. This assumption enables the construction of basic

cryptographic primitives like bit commitment and oblivious transfer with information-theoretic security [111, 97, 72].

- **Entanglement** : By limiting the quantum correlations between adversaries, we can improve the security of certain cryptographic schemes, particularly in the context of position-based cryptography [36]. This is because if the adversaries share an exponentially large number of entangled particles, then any position verification scheme is insecure [33]. However, if we restrict the adversaries to share only a polynomial amount of entangled particles, there exist some schemes [33] for which no attacks are known, although explicit constructions for such schemes are not yet known.
- **Assumption on no-communication** : The no-go theorems state that cryptographic primitives, such as bit-commitment or oblivious transfer, cannot be achieved with information-theoretic security [18, 79, 75]. However, these theorems do not apply in the multi-party setting if we make assumptions about communication between parties. In [18, 85], it was shown that if several spatially separated agents per party are assumed to not communicate, then it is possible to design perfectly secure bit-commitment and oblivious transfer schemes. However, enforcing this non-communication assumption in practice is challenging. In [68], Kent proposed using the special theory of relativity to enforce non-communication between parties, which we will discuss next.
- **Relativistic assumption** : Based on the special theory of relativity and the causality principle, the assumption is made that no physical carrier of information can travel faster than the speed of light. Using this assumption, cryptographic primitives, such as bit-commitment, can be constructed with information-theoretic security [68]. This is achieved by spatially separating the parties in such a way that information cannot be exchanged faster than the speed of light. This model, however, may be difficult to enforce in practice.

Now, we briefly discuss the challenging and fascinating interdisciplinary field called quantum cryptography which is the main focus of this thesis.

1.4 Quantum cryptography

Quantum cryptography aims to make data secure by leveraging the fundamental properties of quantum mechanics such as *entanglement* and *Heisenberg's uncertainty principle*. The main idea behind quantum cryptography is that two parties communicating through a quantum channel can be assured that their communication is not being intercepted by eavesdroppers because measuring a quantum system inevitably disturbs it, and this disturbance will alert the legitimate parties to the presence of eavesdroppers. For this reason, quantum cryptography is considered to be completely secure.

The concept of quantum cryptography was first introduced by Stephen Wiesner in the 1960s when he designed an unforgeable digital banknote using the laws of quantum mechanics. Wiesner also explored the concept of quantum multiplexing

channels [114], where one party could send two messages to another, but the receiver could only read one at the cost of irreversibly destroying the other. Although in back 60's, researchers started to explore the use of quantum mechanics to design cryptographic primitives, the term "Quantum Cryptography" was coined much later by Bennett et al. in 1983 [22]. In [22], the authors pursued the previous ideas of Wiesner to propose the transmission of confidential information over an insecure quantum channel, leading to the first Quantum Key Distribution (QKD) protocol in 1984 [21]. Since its invention in 1983, the field of quantum cryptography has extensively developed in the past few years. Some of its most famous applications are discussed below.

- **Quantum Key Distribution** : Quantum Key Distribution (QKD) is the most well-known and developed application of quantum cryptography which allows two distant parties to communicate securely through an insecure quantum channel. The first QKD protocol was introduced in 1984 by Bennett and Brassard [21] (popularly known as BB84 QKD), but it lacked rigorous security proof. In 2000, Shor and Preskill [103] presented the first complete and simple proof of security for BB84 QKD, and later in 2008, Renato Renner provided a rigorous analysis of security proofs for QKD schemes in his thesis [93]. In 1991, Ekert proposed a QKD scheme based on entanglement and Bell's theorem [46]. Another protocol based on entanglement but not Bell's theorem was presented in 1992 [24]. The first experimental demonstration of QKD was reported in the same year, along with concrete solutions for the classical post-processing phase and security estimates [20]. Since then, quantum cryptography has made significant progress in both theoretical and practical aspects of QKD. A recent article by Ekert and Renner [47] provides an excellent account of the current state of QKD.

Despite quantum key distribution being the main focus of research in quantum cryptography, other applications have also been explored since the early days, beginning with Wiesner's unforgeable quantum money.

- **Quantum Bit Commitment** : Quantum Bit Commitment (QBC) is a cryptographic primitive involving two parties: Alice, who sends a piece of evidence such as a quantum state, and Bob, who receives the evidence. In the commit phase, Alice decides on a value of a bit, either 0 or 1, and sends the evidence to Bob. In the reveal phase, Alice announces the value of the bit and Bob checks it against the evidence. A QBC protocol is considered unconditionally secure if any attempt at cheating can be detected with a probability close to 1. Cheating can occur if Alice tries to change the value of the bit after the commit phase or if Bob tries to learn the value of the bit before the reveal phase.

Bennett and Brassard's original paper on quantum cryptography included a coin-tossing protocol based on bit-commitment [21]. However, the protocol was deemed insecure if one of the parties simply left the quantum states untouched instead of performing the prescribed measurements. This was considered a theoretical threat at that time, given the technical difficulties of implementing such

a strategy. In 1990, Brassard and Crépeau proposed a different quantum bit commitment protocol [30] that was not vulnerable to this issue but was instead vulnerable to an adversary who could perform coherent measurements, i.e., joint measurements on multiple quantum particles, which was also considered difficult. To overcome these limitations, the two protocols were combined to obtain a quantum coin-tossing protocol that can only be broken by an adversary who can both maintain entanglement and perform coherent measurements.

- **Quantum Coin Flipping** : Quantum coin flipping is a distrustful cryptographic primitive that involves two parties communicating through a quantum channel and exchanging information by sending qubits. The first party, Alice, chooses a random sequence of qubits and bases and sends them to the second party, Bob, who records the qubits. Bob then makes a guess about which basis Alice used and reports it back to Alice. Alice then tells Bob whether he guessed correctly or not and sends him her original qubit sequence. Since this is a distrustful primitive, any of the parties may attempt to cheat at any point in the process.

A quantum Oblivious Transfer (OT) protocol was proposed around the same time as the quantum bit commitment protocol in [30] whose security also relies on the assumption that the adversary is limited by technology [23].

Mayers [79] and Lo and Chau [75] independently proved that unconditionally secure quantum bit commitment is impossible. In that same paper [75], Lo and Chau also proved that ideal quantum coin flipping cannot be achieved with unconditional security. Lo further demonstrated the impossibility of unconditionally secure one-out-of-two oblivious transfer and other secure two-party computations [74]. The same techniques used in [74] can be extended to rule out any one-sided two-party computation, where inputs from both parties produce an output that is only given to one of them. The more complicated case of two-sided computation, for a restricted class of functions, was first considered by Colbeck [41], while the general impossibility result was proven by Buhrman, Christandl, and Schaffner [34].

In 2008, Giovannetti et al. proposed a potential solution to the problem of achieving an unconditional secure single-server SPIR scheme by introducing a weaker primitive called Quantum Private Query (QPQ).

- **Quantum Private Query** : Quantum Private Query (QPQ) is a two-party mistrustful cryptographic primitive that offers the same functionality as Symmetric Private Information Retrieval (SPIR) and Oblivious Transfer (OT) but with a weaker security requirement where the user is allowed to get some probabilistic knowledge about her unintended data bits, and the server is allowed to get some information regarding the client's query indices in a cheat sensitive way i.e., if the server tries to retrieve more information about the user's query indices, the user can detect it.

While quantum cryptographic protocols discussed above offer improved security

compared to their classical counterparts, imperfect implementation or faulty devices involved in the protocols can compromise their security. As a result, these protocols are considered “probably secure,” meaning that their security relies on the assumption of perfect operation of the involved devices.

Initially, all quantum cryptographic protocols were proposed assuming perfect devices, which made them device-dependent. However, now researchers are focusing on developing device-independent versions of those schemes, where security does not rely on the trustful assumptions imposed over the quantum devices involved in the scheme. Therefore, the security analysis of such protocols needs to consider scenarios where devices are imperfect or even malicious.

1.5 Device independent quantum cryptography

Even though certain quantum cryptographic primitives offer unconditional security, this does not necessarily mean that their implementations are secure. Any deviation from the protocol’s specifications due to imperfect hardware can lead to side-channel attacks [56, 107], thereby undermining the trust in these systems. To combat this issue, Device-Independent (DI) quantum cryptography has emerged as a solution. These protocols aim to design secure cryptographic schemes that can be implemented with untrusted devices.

The work initiated by Mayers and Yao in [80] laid the foundation for the concept of “self-testing” quantum apparatus. This approach involves utilizing Bell inequalities or non-local games to assess the quantum nature of devices based on their input-output statistics. By examining whether the devices sufficiently violate a Bell inequality, certain properties about the devices can be inferred. Expanding on this idea, Roger Colbeck proposed the use of Bell tests to verify the integrity of devices in his thesis [41]. Over time, notable advancements have been made in the development of protocols that are both unconditionally secure and device-independent for various problems, even when the physical devices employed to conduct the Bell tests are noisy or far from ideal.

Device-independent (DI) cryptography has become a highly active research area within the realm of quantum cryptography. It has seen significant developments, particularly in the context of QKD [7, 92, 109] and randomness expansion or amplification [27, 42, 108]. Building upon the foundational concept introduced in [80], Acín et al. proposed a fully DI-QKD protocol [7]. DI-QKD relies on a crucial assumption that there is no communication between the adversary and the quantum devices. Under this assumption, Vazirani and Vidick provided the first comprehensive proof for a DI-QKD scheme in their work [109]. Ongoing research efforts in the field of QKD are dedicated to developing more practical DI schemes that can effectively operate and retain their functionality even in the presence of realistic levels of noise.

In contrast, two-party cryptography remains an area that has received relatively little exploration within the DI framework. Recent analyses have focused on the security of protocols for imperfect coin flipping and bit commitment in the DI regime [12, 104]. Notably, these works differ in their approach as they do not impose additional

assumptions, leading to the pursuit of imperfect implementations rather than achieving a perfect primitive. Additionally, Adlam and Kent have proposed a DI relativistic bit commitment protocol [9], which offers security for a specific duration under the assumption that each party is divided into agents separated in space-like regions. Furthermore, there have been advancements in the DI scenario for multi-round protocols on bit commitment [12], weak string erasure [66], and single-shot DI settings for weak coin flipping [11] and XOR oblivious transfer [73]. These recent contributions in two-party cryptographic primitives suggest the advancements of the DI scenario beyond QKD.

1.6 Contribution and organization of the thesis

Within the realm of two-party distrustful cryptographic primitives, this thesis specifically centers around the Quantum Private Query (QPQ) primitive in the Device-Independent (DI) scenario. We have already briefly introduced this primitive in Section 1.4, and now, in this section, we will outline our contributions in the subsequent chapters of this thesis, with a specific focus on the field of QPQ in the DI paradigm.

In Chapter 2, we present an overview of the fundamental concepts of quantum mechanics as a foundational understanding for the thesis. This chapter provides a concise discussion of existing results and the mathematical formalisms employed in quantum mechanics, which are utilized for achieving the outcomes mentioned in this thesis.

Next, in Chapter 3, we present an overview of the QPQ primitive, along with a discussion on existing results within this domain. We discuss the relation between QPQ and other related primitives (i.e., SPIR, OT) and also conduct a comparative study with the exact classical counterpart. We also introduce the security definitions that we have defined in our works to analyze the performance of our QPQ proposals and the assumptions taken in our proposals. Notably, Chapter 3 does not present any specific achieved results, apart from the security definitions defined and the assumptions taken in our proposals.

Following the introductory framework established in Chapter 3, the discussion in subsequent chapters (starting from Chapter 4) focuses on the results achieved in this thesis. In Chapter 4, we discuss our proposed novel QPQ scheme, which stands out as the first known (to the best of our knowledge) fully DI-QPQ scheme employing EPR pairs. This proposal incorporates self-testing of shared EPR pairs, self-testing of projective measurement operators, and self-testing of a specific class of POVM operators to certify full DI. Additionally, this chapter formally addresses the security concerns associated with this scheme.

In Chapter 5, we address the limitations of the semi-DI-QPQ scheme proposed in [77] and propose a full DI version of the QPQ scheme [117] by exploiting a proper self-testing mechanism of observables along with the local version of the tilted CHSH game. We compare the performance of this full DI version of the Yang et al. QPQ proposal [117] with our full DI-QPQ scheme mentioned in Chapter 4. Additionally, we propose a full DI version for a modified version of [117], where the client can

retrieve the maximum raw key bits during the oblivious key generation phase.

Chapter 6 focuses on the finite sample analysis for the semi-DI-QPQ proposal in [77]. In this chapter, we present a comparative analysis between the CHSH and the three-party Pseudo Telepathy game to address the reduction in the overall sample size required for the DI certification of the semi-DI-QPQ scheme [77] in a finite sample scenario.

Moving forward to Chapter 7, we address the limitations of existing single-user and multi-user QPQ schemes. In this chapter, we present a semi-DI multi-user QPQ scheme that enables simultaneous retrieval of different items by each user without revealing their respective choices. This proposal allows each user to retrieve optimal raw key bits during the oblivious key generation phase and evaluates the security issues formally.

In Chapter 8, we conclude the thesis by providing a concise summary of our work, highlighting the key contributions and advancements made in the field of QPQ in the DI paradigm. Additionally, we discuss several intriguing open problems that warrant further exploration in the domain of QPQ.

The contribution of this thesis towards the evolution of QPQ is represented in the form of a line diagram in Figure 1-1

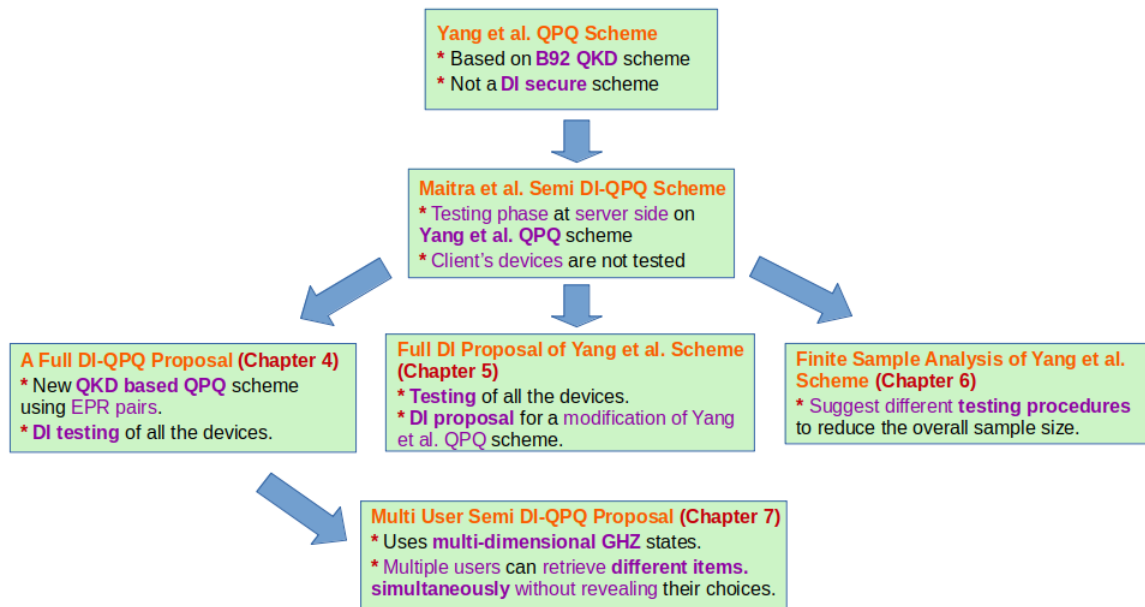


Figure 1-1: Line diagram representing the contribution of this thesis towards the evolution in the field of QPQ

Preliminaries and Background

“I think I can safely say that nobody understands quantum mechanics.”

— Richard P. Feynman, *The Messenger Lectures*, 1964, MIT.

Quantum computing is an interdisciplinary field, encompassing physics, mathematics, and computer science. In this chapter, we review the basic introductory knowledge of quantum mechanics, quantum computation, and quantum cryptography that are relevant to this thesis. Note that most of these contents (mentioned here) about the basics of quantum mechanics, and computation can be found in Nielsen and Chuang’s textbook [86].

Here, we first discuss the mathematical formalism of quantum mechanics: quantum states, measurements, entanglement, distance notion between different states, etc.

2.1 Basics of quantum computation

In this section, we delve into the fundamental concepts and the standard model of quantum computing. Firstly, we introduce the essential notations commonly used in quantum computation, which are summarized in Table 2.1. Subsequently, we explore various topics, including quantum states, measurements, operations, entanglement, and other related background information, in the following subsections.

2.1.1 Quantum bits or qubits

In classical computation and digital communications, the most basic unit of information is represented as a bit, which can be in two states - 0 or 1. Analogously, the equivalent of a bit in quantum mechanical systems is represented as a qubit. Like 0 and 1 bit in classical computation, quantum computation also has its equivalent representation in the form of $|0\rangle$ qubit and $|1\rangle$ qubit respectively. However, unlike

Notation	Description
z^*	Complex conjugate of the complex number z
$ \psi\rangle$	$2^n \times 1$ column vector to represent a n -qubit state. Also known as ket notation.
$\langle\psi $	Dual vector of $ \psi\rangle$. Also known as bra notation.
$\langle\phi \psi\rangle$	Inner product between the vectors $ \phi\rangle$ and $ \psi\rangle$.
$ \phi\rangle \otimes \psi\rangle$	Tensor product between the vectors $ \phi\rangle$ and $ \psi\rangle$.
$ \phi\rangle \psi\rangle$	Abbreviated notation of tensor product between the vectors $ \phi\rangle$ and $ \psi\rangle$.
A^*	Complex conjugate of the A matrix.
A^T	Transpose of the A matrix.
A^\dagger	Hermitian conjugate or adjoint of the A matrix, $A^\dagger = (A^T)^*$
$\langle\phi A \psi\rangle$	Inner product between $ \phi\rangle$ and $A \psi\rangle$. Equivalently inner product between $A^\dagger \phi\rangle$ and $ \psi\rangle$.

Table 2.1: Basic notations used in quantum computation

classical bits, qubits (or quantum bits) can be in a superposition of states (such quantum states are known as “**pure states**”).

Any one qubit pure state $|\psi\rangle$ can be represented as a superposition of the basis states $|0\rangle$ and $|1\rangle$ with certain amplitudes say α and β i.e.,

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle. \quad (2.1)$$

with $\alpha, \beta \in \mathbb{C}$ such that it satisfies the normalization condition $|\alpha|^2 + |\beta|^2 = 1$. That means, any quantum state $|\psi\rangle$ can be fully determined by its two amplitudes α and β and is represented by the complex column vector $\begin{pmatrix} \alpha \\ \beta \end{pmatrix}$ of norm 1. Similarly, the basis states $|0\rangle$ and $|1\rangle$ are represented by the column vectors $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$ respectively.

For any isolated physical system, the state space associated with the system is called the Hilbert space where the system is completely described by its state vector which is a unit vector in the system’s state space. A qubit is described by a two-dimensional Hilbert space (the smallest non-trivial Hilbert space), whose state can take any value of the form mentioned in equation 2.1. Let, Q_1 be the Hilbert space associated with a one qubit pure state. That means, Q_1 can be represented as,

$$Q_1 = \{\alpha, \beta \in \mathbb{C} : |\alpha|^2 + |\beta|^2 = 1\}. \quad (2.2)$$

Here, if one restricts the choice of $\alpha, \beta \in \mathbb{R}$, then a qubit can be represented on a unit circle as depicted in Figure 2-1.

For a *composite quantum system*, the state space is the tensor product of the state spaces of the component physical systems. For example, the state space corresponding to any two-qubit quantum state is $\mathbb{C}^2 \otimes \mathbb{C}^2$, and any of such states can be represented as,

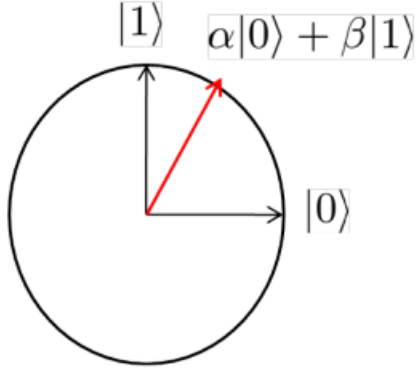


Figure 2-1: Representation of a qubit on a unit circle

$$|\psi_2\rangle = \sum_{i=0}^3 \alpha_i |i\rangle = \alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle.$$

where $\{|i\rangle\}_{i=0}^3$ forms the orthonormal basis in $\mathbb{C}^2 \otimes \mathbb{C}^2$ and α_i 's are complex numbers satisfying the normalization condition $\sum_i \alpha_i = 1$ (here, any two qubit state $|ij\rangle$ actually denotes $|i\rangle \otimes |j\rangle$).

Similarly, any N qubit pure state $|\psi_n\rangle$ can be represented as a superposition of 2^N possible outcomes in $\{0, 1\}^N$ (i.e., the superposition of all possible 2^N basis states) like the following.

$$|\psi_n\rangle = \sum_{i=0}^{2^N-1} \alpha_i |i\rangle = \alpha_0|00\dots 0\rangle + \alpha_1|00\dots 1\rangle + \dots + \alpha_{2^N-1}|11\dots 1\rangle = \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{2^N-1} \end{pmatrix}.$$

In general, any pure quantum state in a d -dimensional Hilbert space $\mathcal{H}_d \simeq \mathbb{C}^d$ can be represented as,

$$|\psi\rangle = \sum_{i=0}^{d-1} \alpha_i |i\rangle. \quad (2.3)$$

where $\{|i\rangle\}_{i=0}^{d-1}$ forms an orthonormal basis for \mathcal{H}^d and $\sum_i |\alpha_i|^2 = 1$.

2.1.2 Operations on qubits

There are only two types of operations that can be performed on a quantum state namely unitary operations and measurements.

- **Unitary Operations** : The evolution of any closed quantum system is described by a unitary operation or equivalently a unitary transformation. Any unitary operator acting on a N -qubit state can be described by a $2^N \times 2^N$ matrix U that satisfies the following condition.

$$UU^\dagger = U^\dagger U = \mathbb{I}.$$

where U^\dagger denotes the conjugate transpose of U and \mathbb{I} denotes the $2^N \times 2^N$ identity matrix. The outcome of a unitary operator U on any pure state $|\psi\rangle$ can be easily determined from the result $U|\psi\rangle$ which is just a standard multiplication of a matrix and a vector.

When a unitary operator is applied to an N -qubit state, it acts on all the superposition states simultaneously. While it is possible to simulate quantum operations using classical computers, this process takes exponential time. This is one of the key reasons why quantum computers are more powerful than their classical counterparts.

Here, we discuss about some of the most commonly used unitary operators or gates in quantum information. At first, we discuss the **single qubit gates** i.e., the unitary operators which act on a single qubit.

- **Identity Operator** : A single qubit identity operator is simply the identity on \mathbb{C}^2 and it's N -qubit generalisation is simply the tensor between N single qubit identity operators $(\mathbb{C}^2)^{\otimes N}$.

- **Pauli Operators** : These are three single qubit operators defined as follows.

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad \sigma_y = -i\sigma_x\sigma_z = \begin{pmatrix} 0 & i \\ -i & 0 \end{pmatrix}.$$

The operator σ_x is known as *bit flip operator* (as it flips a qubit from $|0\rangle$ to $|1\rangle$ and vice versa) and σ_z is known as *phase flip operator* (as it flips the phase whenever the qubit is $|1\rangle$). The operator σ_y performs both bit and phase flip. These operators $\sigma_x, \sigma_y, \sigma_z$ are also denoted as X, Y, Z respectively.

These Pauli matrices generate the Pauli group (denoted by \mathcal{P}_1) with factors $\pm 1, \pm i$ where \mathcal{P}_1 is of the following form.

$$\mathcal{P}_1 = \{\pm I, \pm iI, \pm\sigma_x, \pm i\sigma_x, \pm\sigma_y, \pm i\sigma_y, \pm\sigma_z, \pm i\sigma_z\}.$$

- **Rotation Operators** : The rotation operators or gates represent rotation around different axes. These gates are defined as follows.

$$R_x(\theta) = \begin{pmatrix} \cos \frac{\theta}{2} & -i \sin \frac{\theta}{2} \\ -i \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{pmatrix}, \quad R_y(\theta) = \begin{pmatrix} \cos \frac{\theta}{2} & -\sin \frac{\theta}{2} \\ \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{pmatrix}, \quad R_z(\theta) = \begin{pmatrix} e^{-i\frac{\theta}{2}} & 0 \\ 0 & e^{i\frac{\theta}{2}} \end{pmatrix}.$$

The Pauli operators X, Y and Z can be regarded as special cases of R_x, R_y, R_z respectively with rotation angles of π . The periods of R_x, R_y and R_z are 4π . The rotation operators can be defined in terms of the Pauli operators as follows.

$$R_j(\theta) = e^{\left(\frac{-i\theta A}{2}\right)} = \cos\left(\frac{\theta}{2}\right)I - i \sin\left(\frac{\theta}{2}\right)A, \quad j \in \{x, y, z\}, \quad A \in \{X, Y, Z\}.$$

• **Hadamard Operator** : The Hadamard operator or the Hadamard gate denoted by H is defined as follows.

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

The Hadamard gate acts as the following on the single qubit basis states.

$$H|0\rangle \rightarrow \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right), \quad H|1\rangle \rightarrow \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right).$$

It's N -qubit generalization is denoted as $H^{\otimes N}$.

• **T Gate** : The T gate that operates on a single qubit is defined as follows.

$$T = \begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{i\pi}{4}} \end{pmatrix}.$$

• **Phase Gate** : The phase gate that operates on a single qubit is defined as follows.

$$S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}.$$

Other than these single-qubit operators, some **two-qubit operators** or gates act on two qubits (instead of a single one) and can be defined by a 4×4 unitary matrix.

The most common two-qubit operators are the controlled operators which act on two qubits - a control qubit and a target qubit. Suppose U is an arbitrary single-qubit operation. For any controlled- U (CU) operation, if the control qubit c is set, then U is applied to the target qubit t , otherwise the target qubit t is left alone i.e.,

$$|c\rangle|t\rangle \rightarrow |c\rangle U^c|t\rangle.$$

• **CNOT Operator** : CNOT operator or CNOT gate is a specific type of CU operator with $U = \sigma_x$ (i.e., Pauli X) gate. That means, whenever the control

qubit of the CNOT gate is $|1\rangle$, the target qubit flips, otherwise it remains the same. The operation of the CNOT gate can be defined as follows.

$$|c\rangle|t\rangle \rightarrow |c\rangle X^c|t\rangle = |c\rangle|t \oplus c\rangle.$$

- **SWAP Operator** : SWAP operator or SWAP gate is another popular two-qubit gate that acts on two qubits as follows.

$$\text{SWAP}|\psi\rangle|\phi\rangle = |\phi\rangle|\psi\rangle.$$

The matrix representation of these CNOT and SWAP gates are as follows.

$$\text{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad \text{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Note that the CNOT gate, together with other single-qubit gates, forms a universal set of gates i.e., any N qubit unitary operator can be decomposed as a product of such elementary gates. By restricting the single-qubit gates to Pauli operators and the T-gate, and combining them with the CNOT gate, any N qubit unitary can be approximated to arbitrary precision where the approximation factor ϵ is related to the depth of the underlying quantum circuit according to the result of Solovay and Kitaev [86]. More precisely, this Solovay-Kitaev theorem states that *for any single-qubit gate U and a given accuracy parameter $\epsilon \geq 0$, it is possible to approximate U to the precision ϵ using $O(\log^c(1/\epsilon))$ gates from a fixed finite set, where c is a small constant approximately equals to 2.*

- **Quantum Measurements** : Quantum measurement is described by a collection $\{M_m\}$ of measurement operators that act on the state space of the system being measured. The measurement operators must satisfy the completeness condition i.e.,

$$\sum_m M_m^\dagger M_m = \mathbb{I}.$$

Here, m refers to the measurement outcome generated after the experiment. If a quantum system $|\psi\rangle$ is measured then after measurement, the probability $p(m)$ of occurring the result m is given by,

$$p(m) = \langle \psi | M_m^\dagger M_m | \psi \rangle.$$

After the measurement, the state of the system will be,

$$\frac{M_m|\psi\rangle}{\sqrt{\langle\psi|M_m^\dagger M_m|\psi\rangle}}.$$

For example, one can consider the state $|\psi\rangle$ as mentioned in equation 2.1. If this state $|\psi\rangle$ is measured in $\{|0\rangle, |1\rangle\}$ basis, then the measurement outcome will be $|0\rangle$ with probability $|\alpha|^2$ and $|1\rangle$ with probability $|\beta|^2$.

There are two types of measurement, Projective Measurement and Positive-Operator-Valued Measurement (POVM). A measurement is called projective measurement if the measurement operators $\Pi_m = M_m^\dagger M_m$ satisfy the property $\Pi_m^2 = \Pi_m$. This measurement has the property that performing the same measurement again immediately after the one yields the same result with probability 1.

If the post-measurement state is not of particular interest, then one can perform a more efficient measurement known as POVM. This measurement is described by a set on non-negative operators $\{E_m\}$ such that $\sum_m E_m = \mathbb{I}$ where the index m denotes the measurement outcome. If a quantum state $|\psi\rangle$ is measured then for this measurement, the probability of getting the measurement outcome m is given by,

$$p(m) = \langle\psi|E_m|\psi\rangle.$$

2.1.3 Mixed states

All the operations and measurements are discussed till now considering the states of the form as mentioned in equation 2.1 which is known as **pure state**. However, sometimes it may not be possible to describe the state of a quantum system only using a state of the form as mentioned in equation 2.1. If a quantum system is in a state $\{|\psi_i\rangle\}_{1 \leq i \leq n}$ with probability p_i then the state of that system is called a **mixed state**. A mixed state is represented by a density matrix or a density operator which is a positive semidefinite operator having unit trace and is represented in the following form.

$$\rho = \sum_i p_i |\psi_i\rangle \langle\psi_i|.$$

Note that a pure state can also be represented in a density matrix form. A quantum state ρ is pure if it has rank 1 or equivalently if it satisfies $\text{Tr}[\rho^2] = 1$.

If the evolution of a closed quantum system is described by a unitary U between time t_1 and t_2 then the corresponding density operators ρ_{t_1} and ρ_{t_2} will be related through the following equation.

$$\rho_{t_2} = U \rho_{t_1} U^\dagger.$$

If we perform a measurement defined by the measurement operators $\{M_m\}_m$ on the density operator ρ then the probability of getting the outcome m will be,

$$p(m) = \text{Tr}(M_m^\dagger \rho M_m).$$

and the post measurement state will be,

$$\frac{M_m^\dagger \rho M_m}{\text{Tr}(M_m^\dagger \rho M_m)}.$$

Density operators can also be used to describe any subsystems of a composite system using the reduced density operator. If we have a bipartite physical system in the state ρ_{AB} on $\mathcal{H}_A \otimes \mathcal{H}_B$, then the reduced density operator for system A can be defined as

$$\rho_A = \text{Tr}_B(\rho_{AB}).$$

where Tr_B denotes the partial trace over system B .

2.1.4 Entanglement

Entanglement is a unique feature in quantum mechanics that captures the form of correlation between multiple quantum systems. A state on $\mathcal{H}_A \otimes \mathcal{H}_B$ is called a product state if it can be written of the form $|\phi\rangle \otimes |\psi\rangle$ where $|\phi\rangle \in \mathcal{H}_A$ and $|\psi\rangle \in \mathcal{H}_B$. Any density operator ρ on $\mathcal{H}_A \otimes \mathcal{H}_B$ is called separable if it can be written as a convex combination of such product states. More specifically, we can say that a density operator represents a separable state if it can be written in the following form.

$$\rho = \sum_i p_i |\psi_i\rangle \langle \psi_i| \otimes |\phi_i\rangle \langle \phi_i|. \quad (2.4)$$

where $\{|\psi_i\rangle\}$ denotes the family of states in \mathcal{H}_A and $\{|\phi_i\rangle\}$ denotes the family of states in \mathcal{H}_B with the corresponding probability distribution $\{p_i\}$. Operationally, these kinds of separable (mixed) states can be created using Local Operation and Classical Communication (LOCC).

On the other hand, the states that are not separable are called entangled states. More specifically, the pure states that are not product states i.e., can't be written of the form as mentioned in equation 2.4 are called entangled states. The most common examples of two-qubit entangled states are the Bell states or EPR pairs. These Bell states are also considered as two-qubit maximally entangled states. As entanglement can't be created using LOCC, this resource is exploited for tasks that can't be achieved using classical correlations. Although entanglement doesn't carry any information, it is assumed to provide some inherent communication between the distant parties (who share the entanglement among them) and increase the efficiency of many information processing tasks which is evident from the context of nonlocal games, communication complexity and quantum cryptography. For a detailed review

and applications of quantum entanglement, one may refer to [61].

2.1.5 Distance measures between quantum states

Distance measures are mathematical tools used to compare different aspects of systems, such as their information quantity. In the classical scenario, comparing bit strings is usually straightforward by checking their equality or using the notion of Hamming distance. However, comparing quantum states is more complicated due to the probabilistic nature of measurement and the continuous vector space where the state of a qubit resides. Fortunately, a variety of quantum distance measures have been defined in the literature to handle this problem, each useful for different scenarios in quantum mechanics and quantum information. Here, we only discuss the two notions called Trace Distance (D_{Tr} or Δ) and Fidelity (F) that compare the closeness between two quantum states.

- **Trace Distance:** Trace distance compares two probability distributions p_i and q_i over the same index set as

$$D_{\text{Tr}}(p_i, q_i) = \frac{1}{2} \sum_i |p_i - q_i|$$

- Quantum trace distance measures similarity of two quantum states σ and ρ and is defined as the trace norm of an operator M as,

$$\|M\|_1 = \text{Tr}|M|$$

where $|M| = \sqrt{M^\dagger M}$. For the two quantum states σ and ρ , the trace distance is defined as,

$$\begin{aligned} D_{\text{Tr}}(\sigma, \rho) &= \text{Tr}|\sigma - \rho| \\ &= \|\sigma - \rho\|_1 \end{aligned}$$

where $|A| = \sqrt{A^\dagger A}$ is the positive square root of $\sqrt{A^\dagger A}$.

- **Fidelity:** Fidelity is another measure of the closeness between two probability distributions q_i and p_i , defined as follows,

$$F(q_i, p_i) = \left(\sum_i \sqrt{q_i p_i} \right)^2$$

- For the two quantum states σ and ρ , the fidelity is defined as,

$$F(\sigma, \rho) = \left[\text{Tr}(\sqrt{\sigma^{1/2} \rho \sigma^{1/2}}) \right]^2$$

- The fidelity between pure states $|\phi\rangle$ and $|\psi\rangle$ can be defined as the squared overlap of the states i.e.,

$$F(\sigma, \rho) = |\langle\phi|\psi\rangle|^2$$

where the pure states $|\phi\rangle$ and $|\psi\rangle$ are represented by the density matrix representation $\sigma = |\phi\rangle\langle\phi|$ and $\rho = |\psi\rangle\langle\psi|$ respectively.

- There is a relationship between trace distance and fidelity, two measures of similarity between quantum states, as shown in [49],

$$1 - \sqrt{F(\rho, \sigma)} \leq \frac{1}{2}\text{Tr}|\rho - \sigma| \leq \sqrt{1 - F(\rho, \sigma)}$$

- The trace distance of two quantum states ρ and σ is linked to their distinguishability. If a referee prepares ρ and σ with a probability of $\frac{1}{2}$ each, and another party (Alice) tries to guess which state was prepared, then Alice's optimal probability p_{correct} is linked to the trace distance through,

$$p_{\text{correct}} = \frac{1}{2} \left(1 + \frac{1}{2}\text{Tr}|\rho - \sigma| \right)$$

Thus, the trace distance is proportional to the maximum success probability in identifying the two states. For more information, see [58].

2.1.6 Distinguishability of quantum states

It is well known that because of the no-cloning theorem, it is impossible to retrieve the complete classical description of a quantum system from only a single copy of the state. However, from multiple copies of the same quantum system, it is possible to retrieve the exact description of the system with more certainty. This is the *state estimation problem* in quantum information.

Another related but different problem in this domain is the problem of *state discrimination*. The *state discrimination problem* refers to the problem of identifying or distinguishing an unknown state (pure or mixed) ρ from a set of known states. More specifically, from the set $\{p_i, \rho_i\}_{i=1}^N$ i.e., from the ensemble of states ρ_i 's where each happening with probability p_i , the problem is to identify a particular given state ρ from this set by performing optimal measurement (projective or POVM) that leads to the *minimum error discrimination* probability.

The *Holevo-Helstrom bound* is a well-known result in quantum information theory that determines the following optimal probability for discriminating between two mixed states.

$$\text{Pr}_{\text{guess}}^{\text{opt}} = \frac{1}{2} + \frac{1}{2} \|p_1\rho_1 - p_2\rho_2\|_1 = D_{\text{Tr}}(p_1\rho_1, p_2\rho_2). \quad (2.5)$$

On the other hand, for a pure d -dimensional state $|\psi\rangle$ known to be either $|\psi_1\rangle$ or $|\psi_2\rangle$, there may be two types of errors - 1) the wrong guessing probability p_1

whenever the state is $|\psi_1\rangle$ and 2) the wrong guessing probability p_2 whenever the state is $|\psi_2\rangle$. In this scenario, the following optimal strategy minimizes the parameter “ $\max(p_1, p_2)$ ” with projective measurements known as *Neumark’s measurements* which has an indirect consequence with Neumark’s theorem (or Naimark’s theorem) for general POVMs.

The best discrimination strategy for two pure states $|\psi_1\rangle$ and $|\psi_2\rangle$ with projective measurements $\{ |v_1\rangle, |v_2\rangle \}$, where $|v_1\rangle, |v_2\rangle$ are in the span of $|\psi_1\rangle$ and $|\psi_2\rangle$ such that $\langle v_1 | v_2 \rangle = 0$, they are symmetric with respect to the angle bisector of $|\psi_1\rangle$ and $|\psi_2\rangle$, and $|v_i\rangle$ is closer to $|\psi_i\rangle$ for $i = 1, 2$. Also consider that the angle θ between $|\psi_1\rangle$ and $|\psi_2\rangle$ (detailed orientation in Figure 2-2) is defined as $\theta = \arccos |\langle \psi_1 | \psi_2 \rangle|^2$. In this scenario, if one adopts the strategy that he guesses $|\psi_i\rangle$ whenever the outcome is $|v_i\rangle$, then the success probability is given by,

$$\Pr_{\text{succ}} = |\langle v_i | \psi_i \rangle|^2 = \cos^2 \left(\frac{\pi - \theta}{2} \right) = \frac{1}{2} + \frac{1}{2} \cos \left(\frac{\pi - \theta}{2} \right) = \frac{1}{2} + \frac{1}{2} \sin \theta.$$

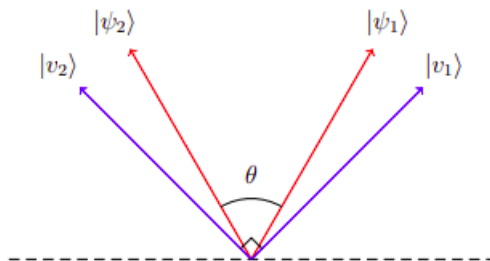


Figure 2-2: Pure states and the corresponding measurement basis for Neumark’s measurement

One can check that in a special case, this optimal probability can be obtained from the Holevo-Helstrom bound [60] as mentioned in equation 2.5.

From the description of the above discrimination strategy using projective measurement, it is clear that some false results occur during the prediction. However, for the scenarios where conclusive outcomes are required, these optimal strategies may not be effective. Fortunately, there exists another discrimination strategy that allows inconclusive outcomes instead of false results. That means if a measurement outcome in this strategy indicates one of the states, then the predicted state will be the correct result with certainty. In the literature, this strategy is known as *Unambiguous State Discrimination (USD)* [62]. This kind of discrimination strategy aims to find the optimal POVM that minimizes the probability of an inconclusive outcome. For the case of two pure states with equal probability of occurrence, the following optimal strategy provides the maximum conclusive success probability of discrimination.

To unambiguously discriminate [62] two pure states say $|\psi_1\rangle$ and $|\psi_2\rangle$, the best strategy is to carry out the POVM $\{D_1, D_2, D_3\}$ where the result D_1 implies that the unknown state is $|\psi_2\rangle$, the result D_2 implies that the unknown state is $|\psi_1\rangle$ and the result D_3 implies that the discrimination is inconclusive. If the angle between the states $|\psi_1\rangle$ and $|\psi_2\rangle$ is θ and the unknown state provided is (say) ψ_1 , then the optimal probabilities of different outcomes for this case will be as follows.

$$\begin{aligned}\Pr(\text{outcome } D_1) &= \text{Tr}(D_1\rho) = 0 \\ \Pr(\text{outcome } D_2) &= \text{Tr}(D_2\rho) = (1 - \cos \theta) \\ \Pr(\text{outcome } D_3) &= \text{Tr}(D_3\rho) = \cos \theta.\end{aligned}$$

Where ρ is the density matrix representation of the unknown state.

The problem of unambiguous state discrimination has been generalized recently to N linearly independent states in [37], and has also been studied for mixed states in [25, 95]. More recent developments on this topic can be found in [67].

In the next chapter, we move our discussion towards the overview of the primitive called *Quantum Private Query (QPQ)*, which is the main area of work in this thesis.

Quantum Private Query

3.1 Overview

The invention of quantum cryptography by Bennett and Brassard [21] in 1984 has led to a surge of interest in secure communication due to its increased security compared to classical cryptography. In addition to secure communication, the use of quantum mechanical properties in cryptography has enabled many functionalities that were previously impossible to achieve classically. One such functionality is Symmetric Private Information Retrieval (SPIR), which is made possible in a single server scenario (in terms of the Quantum Private Query primitive) through the incorporation of quantum mechanics into cryptography.

SPIR is a distrustful primitive involving two parties, where a client requests specific data bits from a server's database without revealing the indices of the requested bits (user privacy), and the server does not disclose any information about the data bits that is not requested (database security). Oblivious Transfer (OT) also provides the same functionality. However, it has been shown in [74] that it is impossible to achieve information-theoretic secure SPIR or OT schemes in a single server scenario without further assumptions. Nonetheless, an unconditional secure SPIR scheme [71] can be designed in a distributed database setting, where the servers share randomness and do not communicate with each other.

The *Quantum Private Query (QPQ)* primitive is a solution to the problem of implementing unconditional secure single server SPIR or OT schemes. QPQ offers the same functionality as SPIR and OT but with a weaker security requirement where the client can know some probabilistic knowledge about her unintended data bits, and the server is assumed not to cheat as there exists a non-zero probability of being caught cheating [4]. If the server tries to retrieve more information about the client's query indices, the client can detect that, and it may ruin the server's reputation as a database owner [63].

3.2 Relation between QPQ, SPIR and OT

Oblivious Transfer (OT) is a well-studied cryptographic primitive that was first introduced informally by Wiesner [114] and then subsequently formalized as 1 out of 2 OT in [101]. In 1 out of N oblivious transfer protocol, a client wants to privately learn one of N entries from a database owned by a server, without the server knowing which entry the client is interested in (known as “user privacy”) and also the client should not know anything about the unintended data bits (known as “data privacy”). This scheme is also referred to as Symmetric PIR (or SPIR). However, there is a minimal difference between SPIR and OT. The main difference between PIR (a weaker version of SPIR where only user privacy is maintained) and OT is that PIR has bound on the communication complexity of the protocol (more specifically, it requires “sublinear” communication in the size of the database) whereas OT has no such requirements. For SPIR, generally, multiple databases are involved to achieve both low communication complexity and information-theoretic security. Although one can have both distributed PIR and distributed OT, because of this communication requirement, PIR is already non-trivial even if the input of the receiver is protected (asymmetric PIR). On the other hand, Symmetric PIR implies 1 out of N OT (but not the other way around).

Alternatively, private query protocols provide similar functionalities to OT and SPIR, but their security requirements are generally relaxed [54]. In the QPQ primitive, it is assumed that the server will not cheat if there exists a non-zero probability of being caught, and the client may obtain a few extra entries, but the number is strictly bounded. All existing QPQ protocols are designed for a single database and are like probabilistic 1 out of N OT or imperfect version of (quantum) SPIR with imperfect data privacy.

It is already shown in [74] that information-theoretically secure two-party computational schemes are impossible in the quantum scenario. This implies the impossibility of designing an information-theoretic secure OT scheme that satisfies both client’s and server’s security requirements. This result also suggested the impossibility of designing information-theoretic secure (quantum or classical) SPIR with a single server without further assumptions. However, the information-theoretic secure single-server classical PIR scheme does exist with $\Theta(n)$ communication complexity [38] (where n is the size of the database). For quantum PIR with a single classical database, Baumeler and Broadbent [17] first come up with an information theoretic secure scheme (with the assumption of specious server) having $\Theta(n)$ communication complexity. Later, Le Gall [50] proposed a scheme with $O(\sqrt{n})$ communication complexity, and Kerenidis et al [69] came out with two proposals - one having $O(\log(n))$ communication with the requirement of linear pre-shared entanglement and the other one having poly logarithmic communication with no requirement of pre-shared entanglement. For multi-server scenario, both classical [53] and quantum [71] SPIR schemes do exist having communication complexity around $O(n^{\frac{1}{2k-1}})$ (where k is the number of servers) with the assumption that the servers don’t communicate with each other. There are also results for quantum PIR with a single quantum database (i.e., a single server holding a database where the database elements are qubits) and quantum SPIR

for multiple quantum databases [105] having both linear and sub-linear communication complexity. However, in this present effort, we are only interested in classical databases.

All the existing literature demonstrates that OT schemes only offer computational security, while SPIR schemes have high storage overhead and impractical assumptions (like no communication between multiple servers) for additional (unconditional) security. Fortunately, due to the relaxed security requirements of QPQ discussed earlier, it is possible to design unconditionally secure QPQ schemes in a single-server scenario with sub-linear ($O(\log(n))$) communication complexity [54]. This makes QPQ schemes more efficient than existing SPIR or OT schemes in terms of practicality, security, storage overhead, and communication complexity. Additionally, QPQ protocols can resist all attacks, including those using quantum resources, whereas classical or even quantum OT protocols may not be able to defend against such attacks. We have demonstrated the relations between QPQ, SPIR, and PIR through a line diagram in Figure 3-1. We have also revisited the existing results for QPQ and (S)PIR protocols for single-server scenarios (in both classical and quantum settings) in Table 3.1 which is already mentioned in [4]. For further details regarding the relation between QPQ and (S)PIR, one may refer to [4].

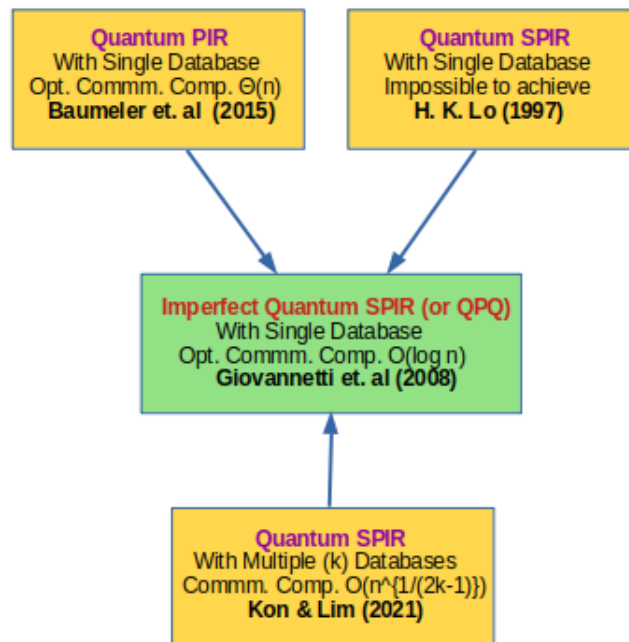


Figure 3-1: Line Diagram for the relations between QPQ, PIR, and SPIR

Problem	Additional Assumption	Opt. Comm. Complexity	Reference
Classical PIR		$\Theta(N)$	[38]
Classical SPIR		NA (Impossible)	
Quantum PIR (Classical Database)	Specious server	$\Theta(N)$	[17]
	Specious server and prior entanglement	$\Theta(N)$	[13]
	Honest server	$O(\text{poly } \log(N))$	[69]
	Honest server and prior entanglement	$O(\log(N))$	[69]
Quantum SPIR (Classical Database)		NA (Impossible)	[74]
	The server will not cheat if there is a non-zero probability of being caught cheating and imperfect data privacy (This is the QPQ primitive).	$O(\log(N))$	[54]
Quantum PIR (Quantum Database)	Honest server and blind setting	$\Theta(N)$	[105]
	Honest server and visible setting	$\Theta(N)$ (for one-round)	[105]
	Honest server and prior entanglement	$O(\log(N))$	[105]
Quantum SPIR (Quantum Database)			

Table 3.1: Known results for (S)PIR protocols in single server scenario

3.3 Comparison with the exact classical primitive

It is well-known that in the classical setting, it is impossible to design an information-theoretically secure OT or SPIR scheme [41]. However, to the best of our knowledge, it is not known whether we can design an unconditional secure *classical private query* (CPQ) scheme. Here, we point out that, it is very easy to come up with a naive and inefficient unconditional secure classical private query scheme. A rough idea of the scheme is given below.

- Suppose, the client Alice wants to know I_1 number of bits from the N bit database X but asks for I_2 positions (that include her I_1 positions) to the server Bob where I_2 is exponentially larger than I_1 but exponentially smaller than N .
- Bob then returns all the bits corresponding to these I_2 positions to Alice. This implies that Alice can't learn more than I_2 bits from the database which is very small compared to the size of the entire database.

- On the other hand, Bob can learn about the positions of Alice’s query with probability $\frac{I_1}{I_2}$ which is also very small.

One can easily check that although this naive classical solution is information-theoretically secure, it has the following disadvantages as compared to the existing quantum solutions.

- In the naive classical solution of the private query primitive, the server Bob leaks more data bits to the client Alice as compared to the existing quantum solutions. In the above-mentioned classical solution, Alice knows an exponential amount of additional data bits as compared to the size of her intended query index set. Whereas, in the quantum scenario, Alice knows a very small amount of additional data bits compared to the size of her intended query index set.
- In the mentioned classical solution, Bob can guess the query indices of Alice with a more certain probability as compared to the existing quantum solutions. In the quantum scenario, Bob guesses each of the data bits as Alice’s query with non-zero probability. Whereas in this mentioned classical solution, Bob can simply eliminate $(N - I_2)$ indices (exponential number of data bits as compared to the size of the query index set) that are not asked by Alice.

The study of designing an efficient classical private query scheme is beyond the scope of the research direction in this thesis.

3.4 Evolution of Quantum Private Query

As discussed in Section 3.2, the concept of designing QPQ protocols emerged as a response to the difficulties encountered in developing unconditionally secure single-server SPIR schemes that enforce a cheat sensitivity in the adversarial model assuming that if there is a non-zero probability of being caught cheating then the server will not cheat.

The history of QPQ protocols began with the proposal of Giovannetti et al. [54]. This was followed by [55] and [88], but these protocols relied on quantum memories, which are not implementable in practice. Several modifications and advancements have been made towards the proposal of the first implementable QPQ scheme by Jakobi et al. [63], which was based on a Quantum Key Distribution (QKD) protocol [96]. This was followed by a flexible generalization by Gao et al.[52] and further efficiency improvements suggested by Rao et al.[91]. Zhang et al.[120] proposed a QPQ protocol based on the counterfactual QKD scheme [87] and Yang et al. developed a flexible QPQ protocol [117] based on the B92 QKD scheme [19]. The domain continues to develop, as seen in recent publications [113, 51]. Based on this discussion of the existing and previous proposals in the field of QPQ, it becomes evident that there are primarily two types of QPQ proposals, namely:

- QPQ protocols based on quantum random access memory (most of the early proposals in this domain) [54, 55, 88].

- QPQ protocols based on quantum oblivious key distribution (most of the recent proposals in this domain) [63, 52, 117].

This thesis is focused on the second type of QPQ schemes (i.e., the QPQ schemes based on QKD). Some of these QKD-based QPQ protocols use entangled states to create a shared key between the server (Bob) and client (Alice), while others use a single qubit sent to the client, which is prepared in specific states and measured to retrieve the key bit. All the recent QKD-based QPQ protocols, despite differences in the key generation procedure, share common concepts. Their security is based on the following fundamental principles.

- The server (Bob) and the client (Alice) share a key between them.
- Bob knows the whole key which would be used for the encryption of the database.
- Alice knows only a fraction of bits of the key.
- Bob does not get any information about the known indices of Alice.

In QPQ, either party may act as an adversary and attempt to compromise security. Alice may try to learn more about the original key bits (that implies the extraction of more data bits in a single query), while Bob may try to learn the indices of the bits known to Alice. Because of this, QPQ is a two-party cryptographic primitive where both parties are distrustful. In reality, the desired primitive is as follows.

- The malicious client Alice’s knowledge of additional data bits is limited to a small fraction beyond what is intended to know by her. The server Bob’s goal is to minimize dishonest Alice’s knowledge of extra information about the database.
- While being honest, the server Bob can only gain limited information about Alice’s query indices. Jakobi et al. [63] demonstrated that dishonest Bob can’t obtain both conclusiveness information and the values of the raw key bits recorded by Alice during the oblivious key generation phase. If Bob attempts to retrieve more information about Alice’s query indices, there is a risk of providing false information about the intended data bits to Alice, which would damage Bob’s reputation as a database owner. Thus, in the QPQ primitive, it is assumed that Bob will not cheat if there exists a non-zero probability of being caught cheating.

Recently, Maitra et al. [77] pointed out that the security of the existing QKD-based QPQ proposals (up until that time) relies on the assumption that the communicating parties trust the devices involved in their scheme, just as it is in the case of initial QKD schemes or for other quantum cryptographic proposals too. In Device Independent (DI) scenario, these trustful assumptions over the devices are removed and security is guaranteed even after removing them. However, unlike QKD, proving DI security for the QPQ distrustful primitive is challenging mainly because of the following reasons.

- In QKD, the parties Alice, and Bob both know all the bits of their shared raw key. However, in QKD-based QPQ schemes, only the server Bob knows all the bits of the shared raw key, and the client Alice knows only some of the shared raw key bits.
- In QKD, both Bob and Alice trust each other, and any third party will act as an adversary. Contrary to this, in QPQ, neither of the parties trusts the other, and any one (or both) of them may act as an adversary.

Despite these challenges, a DI-QPQ scheme has been proposed recently in [77] to enhance the overall security in the QPQ domain by removing trustful assumptions over the devices. However, this protocol only introduced a testing phase on the server side, making it a semi-DI version of the Yang et al. [117] QPQ scheme.

Until that time, there were no full DI proposals in the QPQ domain. In this thesis, we focus on the QPQ domain in the DI paradigm. We come up with some full DI proposals considering different parameters and based on certain assumptions mentioned in Section 3.6. We also analyze the security issues of all our proposals formally by introducing the security definitions discussed next in Section 3.5 of this chapter.

3.5 Security definitions

In the QPQ distrustful cryptographic primitive, none of the parties trusts the other, resulting in different security goals for each party. The security of the entire protocol is termed “Protocol Correctness”, while the security of the server is referred to as “Privacy of the Database Owner” and the security of the client (or each of the clients in the multi-user scenario) is called “Privacy of the User”. To discuss the security issues of our proposals more precisely, we have introduced the following security definitions.

Definition 1. *Protocol Correctness:*

If both the client (or the clients in a multi-user scenario) and the server are honest, then after the protocol execution, it is highly likely that the client (or every client in the multi-user scenario) will correctly retrieve the expected number of data bits in a single database query. That means if the client (or a client in the multi-user scenario) is aware of X data bits and is expected to know Y data bits (according to the scheme), then following the shared key generation phase,

$$\Pr(|X - Y| \leq \delta_t \wedge \text{the scheme doesn't terminate}) \geq P_c. \quad (3.1)$$

where the server tolerates a deviation of δ_t and the probability of X being within the range of $[Y - \delta_t, Y + \delta_t]$ is referred to as P_c , which should ideally be high.

Definition 2. *Protocol Robustness:*

In case of honest implementation of a QPQ scheme, the likelihood of the client (or all of the clients in a multi-user scenario) not knowing any of the final key bits

(or data bits in a single query) and the scheme needing to restart after the shared key generation phase is low. Formally,

$$\Pr(\text{the scheme terminates in honest scenario}) \leq P_a. \quad (3.2)$$

where P_a denotes the probability that no final key bits are known to the client (or any of the clients in a multi-user scenario) and the protocol terminates. Ideally, it should be low.

Definition 3. *Privacy of the Database Owner:*

A QPQ protocol is considered to protect data privacy if, in a single query, a dishonest client (\mathcal{C}^*) can only retrieve at most (on average) τ fraction of bits from the entire N -bit database X , where τ ($0 < \tau < 1$) is very small compared to N , or if the scheme terminates with a high probability in the asymptotic limit. If the number of bits extracted (on average) by the dishonest client (or any of the dishonest clients in the multi-user scenario) in a query is denoted as $D_{\mathcal{C}^*}$, then according to the above definition,

$$E_R(D_{\mathcal{C}^*}) \leq \tau N. \quad (3.3)$$

where τ is a small fraction compared to N and the expectation is calculated over the random coin R utilized in the proposal.

The data privacy against a dishonest client (or the dishonest clients in a multi-user scenario) can also be defined (from the correctness definition) in terms of the success probability in guessing more than the expected number of data bits in a single query. In this notion, after the shared key generation phase, either the scheme terminates with high likelihood (as the limit approaches infinity), or the probability that a dishonest client (\mathcal{C}^*) correctly retrieves more data bits than expected and the protocol doesn't terminate is very low. This means that if the number of data bits known to a dishonest client is represented by X and the expected number is represented by Y , then after the shared key generation phase,

$$\Pr(|X - Y| > \delta_t \wedge \text{the scheme doesn't terminate}) \leq P_d. \quad (3.4)$$

where δ_t represents the allowed deviation by the server from the expected number of data bits and the probability that the actual number of data bits (i.e., X) known to a dishonest client lies outside the range of $[Y - \delta_t, Y + \delta_t]$ is denoted by P_d , which should ideally be very low.

Definition 4. *Privacy of the User:*

A QPQ scheme ensures user privacy if either the dishonest server (\mathcal{S}^*) can accurately identify, on average, at most a small fraction δ of indices from the client's query index set \mathcal{I}_l (or from every i -th client's query index set \mathcal{I}_{l_i} in a multi-user scenario) or the scheme terminates with a high likelihood in the long run. If the dishonest server correctly predicts $l^{\mathcal{S}^*}$ number of indices from the client's query index set \mathcal{I}_l (or from any i -th client's query index set \mathcal{I}_{l_i}) then according to the above definition,

$$E_{R'}(l^{\mathcal{S}^*}) \leq \delta l. \quad (3.5)$$

where the expectation is based on the random coin R' utilized in the proposal (the right-hand side of this inequality will be δl_i in case of every i -th client in the multi-user scenario).

The user privacy against the dishonest server can also be defined in terms of the success probability in guessing a query index correctly from the set of the client's query indices (or from any of the client's query index set in case of a multi-user scenario). In this notion, either the proposal terminates with high likelihood (as the limit approaches infinity), or the probability of the dishonest server (\mathcal{S}^*) accurately guessing a query index from the client's query index set \mathcal{I}_l (or from any of the i -th client's query index set \mathcal{I}_{l_i} in a multi-user scenario) and the protocol not aborting is very low. In other words, if the server guesses an index j from the database and the protocol continues, then the probability of j being in the client's query index set \mathcal{I}_l (or in any of the i -th client's query index set \mathcal{I}_{l_i} in case of a multi-user scenario) is low. *i.e.*,

$$\Pr(\text{Server guesses } j \in \mathcal{I}_l \wedge \text{scheme doesn't terminate}) \leq P_u. \quad (3.6)$$

where P_u represents the probability that j is in \mathcal{I}_l (or in \mathcal{I}_{l_i} for any i -th client in case of a multi-user scenario) and the protocol doesn't terminate (P_u should ideally be very small).

3.6 Security assumptions

The list of assumptions for the security of the QPQ proposals involved in this thesis can be summarized as follows.

1. Devices follow the laws of quantum mechanics *i.e.*, the quantum states and the measurement operators involved in our schemes lead to the observed outcomes via the Born rule.
2. Like the recent DI proposal for oblivious transfer from the bounded-quantum-storage-model and computational assumptions in [31], in this thesis also we assume that for all our schemes, the state generation device and the measurement devices (both at honest and dishonest party's end) are described by a tensor product of Hilbert spaces, one for each device and the devices follow the *i.i.d.* assumption such that each use of a device is independent of the previous use and they behave the same in all trials. This also implies that the statistics of all the rounds are independent and identically distributed (*i.i.d.*) and the devices are memoryless. We also assume that the honest party chooses the inputs randomly and independently for each round.

Note : As QPQ is a distrustful primitive, to detect the fraudulent behavior (if any) of the dishonest party, the *i.i.d.* assumption on the inputs chosen by the honest party seems justified here.

3. The honest party can interact with the unknown devices at his end only by querying the devices with the inputs and getting the corresponding outputs

whereas the dishonest party can manipulate all the devices before the start of the protocol. However, we assume that after the protocol starts, the dishonest party can no longer change this behavior - he cannot manipulate any devices held by the honest party, and also cannot “open up” any devices he possesses at his end (i.e., the dishonest party is also restricted to only supplying the inputs and getting the corresponding outputs from the devices after the start of any of our QPQ proposals). We also assume that the dishonest party processes their data in an *i.i.d.* fashion.

4. Generally, in the Device Independent (DI) scenario, it is assumed that the laboratories of the parties are perfectly secured, i.e., there is no communication between the laboratories. As QPQ is a distrustful primitive, here we assume that each party’s aim is not only to retrieve as much additional information as possible from the other party but also to leak as little additional information as possible from his side. For this reason, while testing the cheating of a dishonest party (or parties) in a particular testing phase, the party who wants to find out the cheating must act honestly in that test to detect the fraudulent behavior (if any) of the dishonest party (or parties). If all the parties act deceitfully in any testing phase, then none of them can detect the cheating of any other party (or parties). So, one party must act honestly in every testing phase.

In the local tests, the honest party performs the test at his end and chooses the input bits randomly for the devices (on behalf of the referee). So, there is no communication between the laboratories. But for distributed tests (i.e., the tests performed by both of them with the shared states), we assume that the honest party chooses the input bits for all the parties on behalf of the referee and then the dishonest party (or parties) announces the measurement outcomes. Therefore, in the case of distributed tests, communication is permitted from the honest party’s end regarding the input bits and from the dishonest party’s (or parties’) output bits.

We also assume that the honest party can somehow “shield” his devices such that no information (regarding the inputs and the outputs) is leaked from his laboratory until he chooses to announce something.

Note : Here, one may think that in the case of a distributed test, the dishonest party (or parties) may not measure his (their) qubits according to the input values chosen by the honest party. In that case, how the honest party can detect this dishonest behavior in the corresponding testing phase is mentioned later in the analysis of *device-independent security*.

5. The inputs for self-tests are chosen freely and independently i.e., the device used to generate input bits for one party does not have any correlations (classical or quantum) with the devices of the other parties involved in a scheme.

Improved and Formal Proposal for Fully Device Independent QPQ using EPR Pairs

Recently, Maitra et al. [77] highlighted that the security of the existing QPQ proposals (up until that time) relied on the assumption of trust in the involved devices, similar to initial QKD proposals. However, in a Device Independent (DI) scenario, these assumptions are removed, and security is guaranteed without relying on trust. Nevertheless, achieving DI security in QPQ is challenging due to its inherent lack of trust between the parties.

Despite this challenge, a DI-QPQ scheme was recently proposed in [77], and its analysis for finite sample scenario was discussed in [16]. However, the proposal in [77] only introduced a testing phase on the server's side, making it a semi-DI version of the QPQ scheme by Yang et al. [117].

Similar to [117], most QKD-based QPQ schemes involve the client generating a partial key by distinguishing non-orthogonal states. In the case of [117], Bob and Alice share non-maximally entangled states, and Alice randomly measures her qubits on a specified basis to retrieve the raw key bits chosen by Bob with certainty.

It is well-known that maximally entangled states are easier to prepare and more robust in a DI setting compared to non-maximally entangled states. Additionally, according to [62, 90], unambiguous discrimination strategy using POVM measurements offers an optimal distinction between non-orthogonal states (discussed in detail in subsection 2.1.6 of Chapter 2).

Taking these factors into consideration, here in this chapter, we propose a new QPQ protocol that utilizes shared EPR pairs and optimal POVM measurements at the client's side to distinguish non-orthogonal quantum states and extract the maximum number of raw key bits during the oblivious key generation phase. This scheme provides full device-independent certification through self-testing of shared EPR states, self-testing of POVM measurements (at the client), and self-testing of projective measurements (at the server). Furthermore, we provide a formal discussion of the security aspects and establish upper limits on the maximum cheating probabilities for both the server and the client.

The chronology of this Chapter can be described as follows. Atfirst, we explain

our exact contributions in detail in Section 4.1. Then we discuss our proposed full DI-QPQ scheme using EPRR pairs in Section 4.2. In Section 4.3, we discuss the security related issues of our proposal formally and derive an upper limit of the maximum cheating probabilities for both the dishonest client and the dishonest server. In the next section (Section 4.4), we discuss the procedure of choosing initial samples in finite sample scenario. Next, in the subsequent sections (Sections 4.5, 4.6), we mention the detailed proofs of our results. At last, in Section 4.7, we discuss the correctness of our proposal considering devices “up to a unitary”.

4.1 Contribution of this chapter

In the chapter, we study the QPQ distrustful primitive in device independent (DI) scenario. Given the distrustful nature of QPQ, proving its DI security is a challenging task. With that in mind, our proposal in this chapter aims to maintain both data privacy and user security, while also detecting any attempts by a party to compromise the system’s security. The main focus of this chapter is outlined below.

1. Unlike the previous semi-DI version in [77], here we come up with a full DI-QPQ scheme using maximally entangled EPR pairs for better preparation and robustness in DI certification. Our proposed QPQ scheme removes device trustworthiness by performing self-testing of EPR pairs (following the procedure of CHSH test), projective measurements operators (following the procedure mentioned in [65]), and POVM operators (following a new strategy mentioned here without imposing any dimension bound). All these self-testing mechanisms provide full DI security, a first of its kind in QPQ (as far as we know). We thoroughly examine the connection of QPQ with comparable primitives, such as OT and SPIR, and compare it with its classical counterpart.
2. Our proposal utilizes optimal POVM measurement at the client’s side, replacing the traditional projective measurement. This allows the client Alice to accurately distinguish two non-orthogonal states with maximum probability, improving the efficiency of the scheme. The result is that, on average, Alice is able to secure the maximum number of raw key bits with certainty. That means, our proposal also enables Alice to retrieve the optimal number of data bits in a single query.
3. We introduce the security definitions for data privacy and user privacy in terms of the maximum fraction of information known by the dishonest party, and also in terms of the probability with which the dishonest party guesses more than the expected amount of information in a scenario where the protocol doesn’t abort. We formally evaluate the performance of our proposal in terms of these security definitions considering all types of attacks that maintain the accuracy requirement. Taking into account all our assumptions, we perform a detailed DI security analysis of our proposal to certify all the devices. We also discuss briefly about the practical implementation (considering finite number of samples) of our scheme.

4.2 Our DI-QPQ proposal

The QPQ protocols are composed of several phases. Depending on the functionality, we have divided the entire protocol into five phases. The first phase is termed the “entanglement distribution phase”. In this phase, a third party (need not be a trusted one and may collude with the dishonest party) distributes several copies of entangled states between the server (Bob) and the client (Alice). The next phase is called the “source device verification phase”. In this phase, the server and the client self-test their shared entangled states using the CHSH game. The third phase is termed as “*Bob’s measurement device verification phase*”. In this phase, Bob self-tests his measurement device (in some specific measurement basis that will be used for the QPQ protocol).

In QPQ, before the protocol, the server Bob decides how much information the client Alice can retrieve from the database in a single query. For this reason, Bob chooses a parameter θ and performs measurements on his qubits (of the shared entangled states) in this θ rotated basis (during the protocol) to restrict Alice’s information about the database ¹. As Alice and Bob get the measurement devices from an untrusted third party, (in the device-independent setting) they need to check the devices before proceeding with the protocol. Here we assume that dishonest Bob’s aim is not only to know Alice’s query indices but also to leak as little additional information about the database as possible. For this reason, in “*Bob’s measurement device verification phase*”, only Bob will act as a referee and choose input bits for both parties. They first perform some measurements assuming the devices as unknown boxes and then after getting the outcome, they conclude about their functionality. After measurement, if the probability of winning the specified game is equal to some predefined value, then they can conclude that Bob’s measurement devices are noiseless for those specified bases.

The next phase of this protocol is termed “*Alice’s POVM device verification phase*”. In this phase, Alice first performs specific measurements assuming the POVM devices as unknown boxes and then concludes about their functionality based on the outcome i.e., in this phase, Alice checks the functionality of her POVM device. If the POVM device works as expected, then Alice and Bob generate key bits in the next phase for the remaining instances which is termed as “*shared key generation phase*”. After this phase, Bob has a secret key such that Alice knows some of those bits and Bob doesn’t know the indices of the bits known by Alice.

In the last phase, i.e., in “*private query phase*”, Bob encrypts the database using the key generated at his side and sends the encrypted database to Alice. Alice then decrypts the intended data bits using the known key bits at her side.

Here we outline the different steps of our proposal in detail. Note that this proposal follows all the assumptions mentioned in Chapter 3 Section 3.6. Also, note that we have not taken into account channel noise and therefore all operations are assumed to be flawless.

¹Once chosen, this value of θ remains fixed for the entire QPQ protocol

1. Entangled State Sharing Phase:

- (a) A third party distributes \mathcal{K} (where \mathcal{K} is assumed to be asymptotically large) number of states, $|\phi\rangle_{\mathcal{A}\mathcal{B}}$, between Alice and Bob with Alice receiving subsystem \mathcal{A} and Bob receiving subsystem \mathcal{B} in each pair.

2. Source Device Certification Phase:

The source device verification phase is composed of two subphases. In the first subphase, Bob acts as a referee, chooses random samples (for testing phase), receives the corresponding qubits from Alice, generates random input bits for those instances and performs a localCHSHtest to certify the states. Similarly, in the second subphase, Alice acts as a referee and does the same that Bob does in the previous phase. In each phase, after receiving the inputs, Alice's and Bob's devices measure the states and return output bits (c_i, b_i) . The detailed description of different subphases is as follows.

- (a) Bob chooses $\frac{\gamma_1 \mathcal{K}}{2}$ instances randomly from these \mathcal{K} shared states (in practice, how Bob and Alice choose the specific value of γ_1 from the set $[0, 1]$ is mentioned in Section 4.4), declares those instances publicly and constructs the set $\Gamma_{\text{CHSH}}^{\mathcal{B}}$ with these chosen instances.
- (b) For all the instances in $\Gamma_{\text{CHSH}}^{\mathcal{B}}$, Alice sends her qubits to Bob.
- (c) For the instances in $\Gamma_{\text{CHSH}}^{\mathcal{B}}$, Bob plays the role of the referee as well as the two players and plays LocalCHSH game.
- (d) For every i -th sample in $\Gamma_{\text{CHSH}}^{\mathcal{B}}$, Bob randomly generates input bits r_i and s_i for his two measurement devices (these devices act as separate parties without any communication), with $r_i, s_i \in \{0, 1\}$.
- (e) Bob performs LocalCHSHtest($\Gamma_{\text{CHSH}}^{\mathcal{B}}$, Bob), according to the procedure outlined in Algorithm 1 (which is equivalent to the local version of the CHSH game) for the set $\Gamma_{\text{CHSH}}^{\mathcal{B}}$.
- (f) If Bob passes this LocalCHSHtest($\Gamma_{\text{CHSH}}^{\mathcal{B}}$, Bob) then both Alice and Bob proceed further, otherwise they abort.
- (g) From the rest $(\mathcal{K} - \frac{\gamma_1 \mathcal{K}}{2})$ shared states, Alice randomly chooses $\frac{\gamma_1 \mathcal{K}}{2}$ instances, declares those instances publicly and constructs the set $\Gamma_{\text{CHSH}}^{\mathcal{A}}$ with these chosen instances.
- (h) For all the instances in $\Gamma_{\text{CHSH}}^{\mathcal{A}}$, Bob sends his qubits to Alice.
- (i) For these instances in $\Gamma_{\text{CHSH}}^{\mathcal{A}}$, Alice plays the role of the referee as well as the two players and plays LocalCHSH game.
- (j) For every i -th sample in $\Gamma_{\text{CHSH}}^{\mathcal{A}}$, Alice randomly generates input bits r_i and s_i for her two measurement devices (these devices act as separate parties without any communication), with $r_i, s_i \in \{0, 1\}$.
- (k) Alice performs LocalCHSHtest($\Gamma_{\text{CHSH}}^{\mathcal{A}}$, Alice), according to the procedure outlined in Algorithm 1 (which is equivalent to the local version of the CHSH game) for the set $\Gamma_{\text{CHSH}}^{\mathcal{A}}$.

Algorithm 1: LocalCHSHtest(\mathcal{S}, \mathcal{P})

- For every $i \in \mathcal{S}$, \mathcal{P} does the following.
 - (a) The device of \mathcal{P} measures on first qubit of the i -th state for inputs $s_i = 0$ and $s_i = 1$ and outputs $c_i = 0$ or $c_i = 1$.
 - (b) The device of \mathcal{P} measures on second qubit of the i -th state for inputs $r_i = 0$ and $r_i = 1$ and outputs $b_i = 0$ or $b_i = 1$.
- From the inputs s_i, r_i and their corresponding outputs c_i, b_i , \mathcal{P} calculates the following quantity.

$$\mathcal{C} = \frac{1}{|\mathcal{S}|} \sum_{i \in \mathcal{S}} \mathcal{C}_i.$$

where \mathcal{C}_i is defined as,

$$\mathcal{C}_i := \begin{cases} 1 & \text{If } s_i r_i = c_i \oplus b_i \\ 0 & \text{otherwise.} \end{cases}$$

- If $\mathcal{C} = \cos^2 \frac{\pi}{8}$ then \mathcal{P} continues with the protocol, otherwise \mathcal{P} aborts the protocol
(In the case of honest implementation, this exact desired value can be obtained for this algorithm and for all the other algorithms mentioned in this thesis using asymptotically large number of samples. However, in practice, with finite number of samples, it is nearly always impossible to exactly match with the desired value of the estimated statistic. Hence, a small deviation from the desired value is allowed in practice. A discussion regarding the variation of the deviation range with the sample size is mentioned later in Section 4.4. However, how the existing security definitions will vary with the noise parameter, is out of the scope of these present works mentioned in this thesis and we will try to explore this issue in our future works).

- (1) If Alice passes the $\text{LocalCHSHtest}(\Gamma_{\text{CHSH}}^A, \text{Alice})$ test then both Bob and Alice proceed to the next phase where Bob self-tests his measurement device, otherwise they abort.

3. Bob's Measurement Device Verification Phase:

Algorithm 2: OBStest(\mathcal{S})

- For every $i \in \mathcal{S}$, Bob and Alice do the following.
 - (a) Bob randomly generates a bit s_i (either 0 or 1) to input into Alice's device and announces the input publicly.
 - (b) Alice measures her share of the i -th state for inputs $s_i = 0$ and $s_i = 1$, and obtained outputs $c_i = 0$ or $c_i = 1$.
 - (c) Bob has already measured his share of the i -th state for inputs $r_i = 0$ and $r_i = 1$, and obtained outputs $b_i = 0$ or $b_i = 1$.
 - (d) Alice and Bob announce their inputs s_i, r_i and their corresponding outputs c_i, b_i .
- Bob and Alice estimate the following quantity from their declared outcomes.

$$\beta = \frac{1}{4} \sum_{s,r,c,b \in \{0,1\}} (-1)^{d_{srcb}} \alpha^{1 \oplus s} \langle \phi_{AB} | A_c^s \otimes B_b^r | \phi_{AB} \rangle.$$

where $\alpha = \frac{(\cos \theta + \sin \theta)}{|\cos \theta - \sin \theta|}$ and d_{srcb} is as follows ,

$$d_{srcb} := \begin{cases} 0 & \text{If } sr = c \oplus b \\ 1 & \text{otherwise.} \end{cases}$$

- If $\beta = \frac{1}{\sqrt{2}|\cos \theta - \sin \theta|}$, then they continue with the protocol, otherwise they abort the protocol.

- (a) In the previous phase (i.e., in *source device certification phase*), Bob and Alice selected a total of $|\Gamma_{\text{CHSH}}|$ samples, with $\Gamma_{\text{CHSH}} = \Gamma_{\text{CHSH}}^A \cup \Gamma_{\text{CHSH}}^B$. For every i -th instance from the remaining $(\mathcal{K} - |\Gamma_{\text{CHSH}}|)$ samples, Bob performs the following.

- Bob generates random bit $r_i \in_R \{0, 1\}$ for every i -th state, as the input of his device. (essentially, these randomly generated bits serve as the initial key bits for Bob, meaning $R_i = r_i$).
- If $r_i = 0$, Bob measures his share of the i -th state using the operator $\{B_0^0, B_1^0\}$ and produces the output bit $b_i = 0$ and $b_i = 1$ respectively.
- If $r_i = 1$, Bob measures his share of the i -th state using the operator $\{B_0^1, B_1^1\}$ and produces the output bit $b_i = 0$ and $b_i = 1$ respectively.

- Bob announces $a_i = 0$ if his device outputs $b_i = 0$, meaning the operator B_0^0 or B_0^1 was applied for the i -th instance.
 - Bob announces $a_i = 1$ if his device outputs $b_i = 1$, meaning the operator B_1^0 or B_1^1 was applied for the i -th instance.
- (b) Bob chooses $\frac{\gamma_2(\mathcal{K}-|\Gamma_{\text{CHSH}}|)}{2}$ instances randomly from these $(\mathcal{K} - |\Gamma_{\text{CHSH}}|)$ shared states, declares them publicly and constructs a set $\Gamma_{\text{obs}}^{\mathcal{B}}$ with these instances.
- (c) Alice then chooses $\frac{\gamma_2(\mathcal{K}-|\Gamma_{\text{CHSH}}|)}{2}$ instances randomly from the rest $(\mathcal{K} - |\Gamma_{\text{CHSH}}| - \frac{\gamma_2(\mathcal{K}-|\Gamma_{\text{CHSH}}|)}{2})$ shared states, declares the instances publicly and constructs a set $\Gamma_{\text{obs}}^{\mathcal{A}}$ with these instances.
- (d) Bob and Alice create a set Γ_{obs} with all their chosen samples i.e., $\Gamma_{\text{obs}} = \Gamma_{\text{obs}}^{\mathcal{A}} \cup \Gamma_{\text{obs}}^{\mathcal{B}}$.
- (e) They then perform $\text{OBStest}(\Gamma_{\text{obs}})$, by following the procedure mentioned in Algorithm 2, for the set Γ_{obs} .

4. Alice's POVM Device Verification Phase:

- (a) After *Bob's measurement device verification phase*, Alice and Bob move on to this phase with the remaining $(\mathcal{K} - |\Gamma_{\text{CHSH}}| - |\Gamma_{\text{obs}}|)$ shared states, referred to as Γ_{POVM} .
- (b) Alice randomly selects $\gamma_3|\Gamma_{\text{POVM}}|$ samples from Γ_{POVM} , calls this set $\Gamma_{\text{POVM}}^{\text{test}}$, and declares the instances.
- (c) Bob first declares the x values for each of the instances in the set $\Gamma_{\text{POVM}}^{\text{test}}$, and then Alice performs $\text{KEYgen}(\Gamma_{\text{POVM}}^{\text{test}})$ followed by $\text{POVMtest}(\Gamma_{\text{POVM}}^{\text{test}})$ according to the procedures described in Algorithms 3 and 4 respectively for the same set.

Algorithm 3: $\text{KEYgen}(\mathcal{S})$
<ul style="list-style-type: none"> • For every $i \in \mathcal{S}$, Alice performs the following. <ul style="list-style-type: none"> (a) If Bob stated $a_i = 0$, Alice uses measurement device $M^0 = \{M_0^0, M_1^0, M_2^0\}$ to measure her qubit in the shared state indexed by i. (b) If Bob stated $a_i = 1$, Alice uses measurement device $M^1 = \{M_0^1, M_1^1, M_2^1\}$ to measure her qubit in the shared state indexed by i.

Algorithm 4: POVMtest(\mathcal{S})

- In this step, Alice first separates instances where Bob declared $a_i = 0$ into a set \mathcal{S}^0 , and the rest (where Bob declared $a_i = 1$) into \mathcal{S}^1 .
- Alice assumes that for each set \mathcal{S}^y (where $y = a_i$, the values declared by Bob), the states at her side are either ρ_x^y or $\rho_{x\oplus 1}^y$ (where $x = r_i$, the raw key bit values randomly chosen by Bob).
- For each set, Alice calculates the parameter Ω^y as

$$\Omega^y = \sum_{b,x \in \{0,1\}} (-1)^{b \oplus x} \text{Tr}[M_b^y \rho_x^y].$$

where M_b^y is Alice's measurement outcome in KEYgen().

- If for every \mathcal{S}^y ($y \in \{0,1\}$),

$$\Omega^y = \frac{2 \sin^2 \theta}{(1 + \cos \theta)}$$

then Alice continues with the scheme, otherwise Alice aborts the scheme.

5. Key Generation Phase:

- After Alice's POVM device verification phase, Alice continues with the remaining shared states ($|\Gamma_{\text{POVM}}| - \gamma_3 |\Gamma_{\text{POVM}}|$), which she denotes as Γ_{Key} .
- Alice performs the KEYgen (Γ_{Key}) for these shared states.
- After KEYgen(Γ_{Key}), Alice determines the original raw key bits based on her measurement results-
 - For each shared state with $a_i = 0$, if Alice gets $M_0^0(M_1^0)$, she concludes the i -th raw key bit as 0(1). If she receives M_2^0 , she ignores it.
 - Similarly, for each shared state with $a_i = 1$, if Alice obtains $M_0^1(M_1^1)$, she concludes the i -th raw key bit as 0(1). If she receives M_2^1 , she ignores it.
- Bob and Alice then proceed to the private query phase with the shared states in Γ_{Key} . This set contains kN many states, where $k > 1$ and k is exponentially smaller than N , the number of bits in the database.
- Alice and Bob conduct the next phase using the kN raw key bits obtained from the shared states.

6. Private Query Phase:

- Bob and Alice possess a raw key of length kN bits with Bob aware of all its values and Alice aware of some of them (without Bob knowing which bits Alice knows).

- (b) Bob rearranges the order of the kN -bit string by randomly announcing a permutation, and both parties then apply that permutation to their raw key bits.
- (c) Bob divides the raw key into N partitions, each with k bits, and informs Alice of each bit's position. Alice and Bob then XORed the bits of each substring to form the final key, which is N bits long. If Alice is not aware of any bits of the final key, the protocol must be repeated.
- (d) Alice, who knows only the i -th bit of Bob's final key F , requests the j -th bit of the database m_j by announcing a permutation P_A . This permutation moves the i -th bit of the final key to the j -th position. Bob applies the permutation P_A on the final key F and uses it to encrypt the database with a one-time pad. Alice can recover m_j as it is encrypted by F_i after receiving the encrypted database.
- (e) Alice must announce the permutation l times if she wants to retrieve l bits of the database with only one known final key bit.
- (f) If Alice knows more than one final key bit then she announces a permutation that links her known final key bits to the database bits she wants to know. This way she can retrieve multiple intended database bits in a single query.

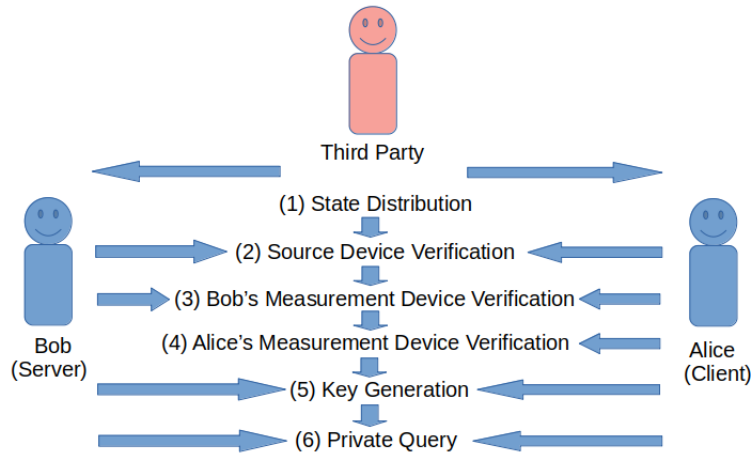


Figure 4-1: Visual representation of different steps of our DI-QPQ proposal.

Our QPQ Proposal (In Case of Honest Implementation)

- Alice and Bob share \mathcal{K} EPR pairs, with Alice having the first qubit and Bob having the second qubit.
- For each shared state, they generate raw key bits by following the procedure mentioned below.
 - Bob randomly generates a value of either 0 or 1 for the i -th raw key bit r_i .
 - If $r_i = 0$, Bob performs measurement on his share for the i -th state in $\{|0\rangle, |1\rangle\}$ basis, otherwise (i.e., for $r_i = 1$) he performs measurement in $\{|0'\rangle, |1'\rangle\}$ basis where $|0'\rangle = (\cos\theta|0\rangle + \sin\theta|1\rangle)$ and $|1'\rangle = (\sin\theta|0\rangle - \cos\theta|1\rangle)$ (the value of θ is chosen as per the relation specified in equation 4.7).
 - Bob announces $a_i = 0(a_i = 1)$ if the outcome at his side corresponding to the i -th shared state is either $|0\rangle(|1\rangle)$ or $|0'\rangle(|1'\rangle)$.
 - When Bob declares $a_i = 0$, Alice performs measurement on her share of the i -th state using the POVM $M^0 = \{M_0^0, M_1^0, M_2^0\}$ where

$$\begin{aligned}
 M_0^0 &\equiv \frac{(\sin\theta|0\rangle - \cos\theta|1\rangle)(\sin\theta\langle 0| - \cos\theta\langle 1|)}{1 + \cos\theta} \\
 M_1^0 &\equiv \frac{(|1\rangle\langle 1|)}{1 + \cos\theta} \\
 M_2^0 &\equiv I - M_0^0 - M_1^0
 \end{aligned}$$

- Similarly, for $a_i = 1$, Alice performs measurement on her share of the i -th state using the POVM $M^1 = \{M_0^1, M_1^1, M_2^1\}$ where

$$\begin{aligned}
 M_0^1 &\equiv \frac{(\cos\theta|0\rangle + \sin\theta|1\rangle)(\cos\theta\langle 0| + \sin\theta\langle 1|)}{1 + \cos\theta} \\
 M_1^1 &\equiv \frac{(|0\rangle\langle 0|)}{1 + \cos\theta} \\
 M_2^1 &\equiv I - M_0^1 - M_1^1
 \end{aligned}$$

- When Bob declares $a_i = 0$, if Alice gets $M_0^0(M_1^0)$, she concludes the i -th raw key bit as 0(1). For measurement outcome M_2^0 , Alice remains uncertain.
- When Bob declares $a_i = 1$, if Alice gets $M_0^1(M_1^1)$ for $a_i = 1$, she concludes the i -th raw key bit as 0(1). For measurement outcome M_2^1 , Alice remains uncertain.

- Atfirst, Bob decides the value of θ and the number of raw key bits needed to generate each bit of the final key based on equation 4.7. Bob and Alice generate a final key by processing their raw key bits through permutation and XOR such that the final key and the database are of the equal size and Bob knows all the bits but Alice knows only some bits of the final key.
- Bob encrypts the whole database using one time pad with his final key and sends it to Alice.
- Alice recovers the desired bits from the encrypted database using her partial knowledge of the final key.

4.3 Analysis of the protocol

In this section, we cover the workings of our proposed scheme. We start by examining the accuracy of the protocol, followed by estimating the security parameters involved. Finally, we delve into the security aspects of our scheme. It's worth noting that all our analyses are based on asymptotic scenarios, and the actual values of parameters may vary in practice based on the sample size selected.

4.3.1 Correctness of the protocol

We begin by demonstrating the accuracy of the protocol.

Theorem 1. *In case of honest implementation of our proposal, on average, Alice can correctly retrieve around $(1 - \cos \theta)kN$ bits of the raw key R at the end of shared key generation phase.*

Proof. Bob and Alice have kN raw key bits after *shared key generation phase*. These raw key bits were generated from kN copies of maximally entangled states of the form

$$\begin{aligned} & \frac{1}{\sqrt{2}}(|0\rangle_{\mathcal{A}}|0\rangle_{\mathcal{B}} + |1\rangle_{\mathcal{A}}|1\rangle_{\mathcal{B}}) \\ &= \frac{1}{\sqrt{2}}(|0'\rangle_{\mathcal{A}}|0'\rangle_{\mathcal{B}} + |1'\rangle_{\mathcal{A}}|1'\rangle_{\mathcal{B}}), \end{aligned}$$

where, $|0'\rangle = (\cos \theta|0\rangle + \sin \theta|1\rangle)$ and $|1'\rangle = (\sin \theta|0\rangle - \cos \theta|1\rangle)$. Here θ may vary from 0 to $\frac{\pi}{2}$. The generation of such kN raw key bits can be redefined as follows.

Bob prepares a random bit stream $R = r_1 \dots r_{kN}$ of length kN . If $r_i = 0$, Bob measures his qubits in $\{|0\rangle, |1\rangle\}$ basis. Whereas, if $r_i = 1$, Bob measures his qubit in $\{|0'\rangle, |1'\rangle\}$ basis. After each measurement Bob announces a bit $a_i \in \{0, 1\}$. If he gets $|0\rangle$ or $|0'\rangle$, he announces $a_i = 0$. If he gets $|1\rangle$ or $|1'\rangle$, he announces $a_i = 1$. Now, Alice's job is to guess the value of each r_i .

Thus, whenever Bob declares $a_i = 0$, Alice can understand that Bob gets either $|0\rangle$ or $|0'\rangle$ and the shared qubit of her side also collapses to $|0\rangle$ or $|0'\rangle$ respectively. However, to obtain the value of the raw key bit, Alice has to distinguish these two states with certainty. As, $|0\rangle$ and $|0'\rangle$ are non-orthogonal states (when $\theta \neq \frac{\pi}{2}$), Alice cannot distinguish these two states with certainty for all the instances.

According to the strategy mentioned in the protocol, whenever Bob declares $a_i = 0$, Alice chooses the POVM $\{M_0^0, M_1^0, M_2^0\}$. After measurement, if Alice receives the outcome M_0^0 , she concludes that Bob's measurement outcome was $|0\rangle$. In such case, Alice concludes that $r_i = 0$. If Alice receives the outcome M_1^0 , she concludes that Bob's measurement outcome was $|0'\rangle$. In such a case, Alice concludes that $r_i = 1$. However, if the measurement outcome is M_2^0 , then Alice remains uncertain about the value of the raw key bit. Alice follows the similar methodology for $a_i = 1$.

Now, we calculate the success probability of Alice to guess each r_i correctly. Let us assume that $\Pr(M_j^{a_i}|\phi_i^{a_i})$ denotes the corresponding success probability of getting the result $M_j^{a_i}$ when the given state is $|\phi_i^{a_i}\rangle$ i.e.,

$$\Pr(M_j^{a_i}|\phi_i^{a_i}) = \langle \phi_i^{a_i} | M_j^{a_i} | \phi_i^{a_i} \rangle.$$

We now calculate the corresponding success probabilities of getting different results for the states $|0\rangle$ and $|0'\rangle$. For $|0\rangle$, the success probabilities will be

$$\begin{aligned} \Pr(M_0^0|0) &= \langle 0 | M_0^0 | 0 \rangle \\ &= (1 - \cos \theta) \\ \Pr(M_1^0|0) &= \langle 0 | M_1^0 | 0 \rangle \\ &= 0 \\ \Pr(M_2^0|0) &= \langle 0 | M_2^0 | 0 \rangle \\ &= \cos \theta \end{aligned}$$

Similarly, for the state $|0'\rangle$, the success probabilities will be

$$\begin{aligned} \Pr(M_0^0|0') &= \langle 0' | M_0^0 | 0' \rangle \\ &= 0 \\ \Pr(M_1^0|0') &= \langle 0' | M_1^0 | 0' \rangle \\ &= (1 - \cos \theta) \\ \Pr(M_2^0|0') &= \langle 0' | M_2^0 | 0' \rangle \\ &= \cos \theta \end{aligned}$$

Similarly, whenever Bob declares $a_i = 1$, Alice chooses the POVM $\{M_0^1, M_1^1, M_2^1\}$. In a similar way, we can calculate the success probability here. We formalize all the conditional probabilities in the following table.

		Cond. Probability of Alice		
		Alice Bob	A= M_0^0/M_0^1	A= M_1^0/M_1^1
0	$B = 0\rangle$	$1 - \cos \theta$	0	$\cos \theta$
0	$B = 0'\rangle$	0	$1 - \cos \theta$	$\cos \theta$
1	$B = 1\rangle$	$1 - \cos \theta$	0	$\cos \theta$
1	$B = 1'\rangle$	0	$1 - \cos \theta$	$\cos \theta$

According to the protocol, if $a_i = 0$ and Alice gets $M_0^0(M_1^0)$, she outputs $r_{\mathcal{A}_i} = 0(1)$. When $a_i = 1$ and she gets $M_0^1(M_1^1)$, she outputs $r_{\mathcal{A}_i} = 0(1)$. Thus, the success probability of Alice to guess the i -th raw key bit r_i of Bob can be written as

$$\begin{aligned}
& \Pr(r_{\mathcal{A}_i} = r_i) \\
&= \Pr(r_{\mathcal{A}_i} = 0, r_i = 0) + \Pr(r_{\mathcal{A}_i} = 1, r_i = 1) \\
&= (1 - \cos \theta).
\end{aligned}$$

So, according to the proposed scheme, the overall success probability of Alice in guessing a raw key bit is equal to $(1 - \cos \theta)$. This implies that at the end of the key establishment phase, Alice can guess (on average) around $(1 - \cos \theta)kN$ many raw key bits with certainty. \square

4.3.2 Estimation of parameters for private query phase

In this subsection, the different parameter values are calculated to ensure that both user and data privacy are preserved. After *shared key generation phase*, Bob and Alice share kN raw key bits, with Bob having full knowledge of them and Alice having partial knowledge. In *private query phase*, both Alice and Bob cut their raw keys in some particular positions to prepare N substrings of length k such that $k = \frac{|\Gamma_{\text{Key}}|}{N}$ where $|\Gamma_{\text{Key}}|$ denotes the total number of raw key bits at the *private query phase* and N denotes the number of database bits. Alice and Bob then perform bit wise XOR among the bits of each substring to get the N bit final key F . Here, $r_i(1 \leq i \leq kN)$ denotes the i -th raw key of Bob and $f_i(1 \leq i \leq N)$ denotes the i -th final key of Bob. Based on the procedure mentioned in *private query phase* for generating final key bits, the relation between r_i and f_i can be written as,

$$f_i = \bigoplus_{j=(i-1)k+1}^{ik} r_j \quad (1 \leq i \leq N)$$

where \oplus denotes addition modulo 2.

It will be clearer by a toy example. Consider $N = 10$ and $k = 2$. Let us assume that the raw key at Bob's side is,

01 10 01 00 10 01 01 11 00 11

and after the *shared key generation phase*, the raw key at Alice's side is,

?1 ?? 0? ?? ?? 01 ?1 ?? 0? ?1

i.e., Alice knows the values of 2nd, 5th, 11th, 12th, 14th, 17th and 20th key bits of the original raw key (? stands for inconclusive key bit i.e., the positions where Alice can't guess the key bits with certainty).

Now, after the modulo operation on the raw key, Bob's final key will be,

$$1\ 1\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 0$$

and Alice's final key will be,

$$?\ ?\ ?\ ?\ ?\ 1\ ?\ ?\ ?\ ?$$

Thus, the number of known key bits by Alice is reduced from 7 to 1. The significance of such modulo operation is to enhance the security of the protocol. This is similar to the privacy amplification in a QKD protocol.

Estimation of the security parameter θ :

In the proposed scheme, Alice can expect to know approximately $(1 - \cos \theta)kN$ of the kN shared raw key bits. The expected value of n_r , the number of raw key bits known to the client Alice after the *shared key generation phase* of our scheme, can be expressed as,

$$E[n_r] = (1 - \cos \theta) kN. \tag{4.1}$$

Bob and Alice combine k raw key bits using XOR to produce each bit of the final key. So, for Alice to correctly guess a final key bit, she must correctly guess all k corresponding raw key bits, which has a probability of $(1 - \cos \theta)^k$. Let n_f denotes the total final key bits known to Alice. It follows that n_f is a binomial random variable with N trials and probability $P_f = (1 - \cos \theta)^k$. Hence, the expected number of final key bits known by Alice after the *shared key generation phase* is,

$$E[n_f] = P_f N \approx (1 - \cos \theta)^k N. \tag{4.2}$$

Our DI proposal requires that dishonest Alice correctly measure (using the designated POVM) in order to pass the DI testing stage. It is recognized that the maximum probability of distinguishing between two non-orthogonal states is $(1 - \cos \theta)$ [62]. This means that in the non-abort scenario, dishonest Alice's guess for the i -th raw key bit, R_i , is limited to at most $(1 - \cos \theta)$ i.e.,

$$\Pr[R_{\mathcal{A}_i^*} = R_i] \leq (1 - \cos \theta). \tag{4.3}$$

where \mathcal{A}_i^* denotes dishonest Alice's subsystem corresponding to the i -th shared state.

As after Bob's measurement, Alice's states are independent and the measurement

devices at dishonest Alice's side are also independent and memoryless, the maximum probability that dishonest Alice can guess the i -th final key bit F_i will be $(1 - \cos \theta)^k$ i.e.,

$$\Pr[F_{\mathcal{A}_i^*} = F_i] = P_f \leq (1 - \cos \theta)^k. \quad (4.4)$$

Based on the results in equation 4.2 and equation 4.4, it can be concluded that in non-abort scenario, the expected maximum number of final key bits guessed correctly by dishonest Alice will be limited by,

$$E[F_{\mathcal{A}^*}] \leq (1 - \cos \theta)^k N. \quad (4.5)$$

Our proposal involves encrypting the database, which is the same size as the final key, by bitwise XORing it with the final key. Thus, a correct guess of a final key bit also implies a correct guess of the corresponding database bit. Hence, in a single query, if the scheme doesn't terminate, dishonest Alice's expected number of correctly guessed database bits is also upper bounded by $(1 - \cos \theta)^k N$ i.e.,

$$E[D_{\mathcal{A}^*}] \leq (1 - \cos \theta)^k N. \quad (4.6)$$

In our scheme, for the protocol to continue, Alice must know at least one final key bit, while Bob wants Alice to know less than two final key bits. Thus, the following condition must be met in the non-abort scenario.

$$1 \leq E[n_f] < 2.$$

This implies that,

$$\begin{aligned} 1 &\leq (1 - \cos \theta)^k N < 2 \\ \frac{1}{N} &\leq (1 - \cos \theta)^k < \frac{2}{N}. \end{aligned} \quad (4.7)$$

All these results boil down to the following conclusion.

Corollary 1. *To ensure that Alice knows atleast one final key bit but no more than one, Bob needs to select k and θ such that,*

$$\frac{1}{N} \leq (1 - \cos \theta)^k < \frac{2}{N}.$$

Now, for our proposal, we derive the limits on the values of P_a (from definition 2) and P_c (from definition 1) set by the correctness condition.

Estimation of the security parameters P_a and P_c :

Initially, we evaluate the likelihood that the protocol won't end during the honest scenario. Then, using the obtained upper bound on $(1 - \cos \theta)^k$ from equation 4.7, we can calculate a lower bound on P_c with the Chernoff-Hoeffding inequality [59] (since we consider a scenario where dishonest Alice measures *i.i.d.*).

Our scheme calculates the probability of Alice correctly guessing a final key bit as $(1 - \cos \theta)^k$. So, the probability of Alice not guessing a final key bit is $\left[1 - (1 - \cos \theta)^k\right]$.

Therefore, the likelihood of Alice not knowing any of the N final key bits is

$$\left[1 - (1 - \cos \theta)^k\right]^N \approx e^{-(1 - \cos \theta)^k N}. \quad (4.8)$$

i.e., for our proposal, we obtain the following bound on the value of P_a .

$$P_a \leq e^{-(1 - \cos \theta)^k N}. \quad (4.9)$$

If Bob sets θ such that $(1 - \cos \theta)^k = \frac{1}{N}$, then equation 4.9 gives us the following result according to the relation in equation 4.7.

$$\boxed{P_a \leq e^{-1}}. \quad (4.10)$$

That means our proposed scheme results in a small value of P_a . The likelihood of our proposal not terminating in an honest scenario, where Alice knows at least one final key bit, is calculated as

$$\Pr(\text{the scheme doesn't terminate}) \geq [1 - e^{-1}]. \quad (4.11)$$

Therefore, our proposed scheme has a high likelihood of not aborting, as demonstrated by the above calculation. We now mention the Chernoff-Hoeffding inequality [59].

Proposition 1. (*Chernoff-Hoeffding Inequality*) Let $X = \frac{1}{m} \sum_{1 \leq i \leq m} X_i$ be the average of m independent random variables X_1, X_2, \dots, X_m with values $(0, 1)$, and let $\mathbb{E}[X] = \frac{1}{m} \sum_{1 \leq i \leq m} \mathbb{E}[X_i]$ be the expected value of X . Then for any $\delta_{CH} > 0$, we have $\Pr[|X - \mathbb{E}[X]| \geq \delta_{CH}] \leq \exp(-2\delta_{CH}^2 m)$.

Our scheme defines $X_i = 1$ if the i -th final key bit is known to Alice (i.e., she gets conclusive outcomes from the POVMs), and $X_i = 0$ otherwise. Total N final key bits result in the random variable X as the sum of these X_i values. If the scheme doesn't terminate, the expected number of final key bits that Alice should know is $Y = (1 - \cos \theta)^k N$.

To ensure the value of X lies within the error margin of $\delta_{CH} = \epsilon(1 - \cos \theta)^k N$ from the expected value, we use the Chernoff-Hoeffding inequality. This is because Alice's final key bits are independent, as the collapsed states and measurement devices are also independent and memoryless. The values of X and Y are calculated under

the assumption that the scheme doesn't terminate. So, using the expression for the Chernoff-Hoeffding bound from proposition 1, we can write that,

$$\begin{aligned} & \Pr [|X - Y| < \delta_{CH} \wedge \text{scheme doesn't terminate}] \\ & \geq 1 - \exp(-2\delta_{CH}^2 m). \end{aligned} \quad (4.12)$$

The *shared key generation phase* results in N final key bits shared by Alice and Bob. Among those N bits, we aim to have the number of final key bits known to Alice fall within $[p - \epsilon p, p + \epsilon p]$, where $p = (1 - \cos \theta)^k N$ and the allowed deviation is $\delta_{CH} = \epsilon (1 - \cos \theta)^k N$. Using the expression in 4.12 with δ_{CH} and m values, we get,

$$\boxed{\begin{aligned} & \Pr [|X - Y| < \delta_{CH} \wedge \text{scheme doesn't terminate}] \\ & \geq 1 - \exp(-2\delta_{CH}^2 N) \\ & \text{where } \delta_{CH} = \epsilon (1 - \cos \theta)^k N \end{aligned}}. \quad (4.13)$$

We have already established the bound $\frac{1}{N} \leq (1 - \cos \theta)^k < \frac{2}{N}$ for $(1 - \cos \theta)^k$ from equation 4.7 for our proposed scheme. If we let Bob choose θ and k such that $(1 - \cos \theta)^k = \frac{1}{N}$, then substituting this value into equation 4.13 will yield,

$$\boxed{\begin{aligned} & \Pr [|X - Y| < \epsilon \wedge \text{scheme doesn't terminate}] \\ & \geq 1 - \exp(-2\epsilon^2 N) \end{aligned}}. \quad (4.14)$$

In our proposal, a correct guess of a final key bit means a correct guess of the related data bit. So, as per definition 1, we can say that when Alice and Bob are both honest, the lower bound of the parameter P_c in our proposal is determined by,

$$\boxed{P_c \geq [1 - \exp(-2\epsilon^2 N)]}. \quad (4.15)$$

That means the likelihood of Alice knowing the expected number of final key bits and the scheme not terminating is high in the honest scenario of our proposed scheme, as the value of N is large in practice.

Bob chooses θ and k so that Alice knows at least one and fewer than two final key bits. So, the deviation, δ_{CH} , has the following bound according to equation 4.7.

$$\epsilon \leq \delta_{CH} < 2\epsilon. \quad (4.16)$$

That means the upper bound of ϵ can be derived from $2\epsilon \leq 1$, yielding $\epsilon \leq \frac{1}{2}$.

To evaluate performance, we consider the variant 1-out-of-2 probabilistic oblivious transfer (for $N = 2$ and $k = 1$). If Bob chooses θ such that $(1 - \cos \theta) = \frac{1}{2}$ (minimum value for $N = 2, k = 1$), the expected number of final key bits (or data bits) Alice can retrieve in a single query is $(\frac{1}{2} \times 2) = 1$. From equation 4.11, we can say that in honest scenario, for this 1 out of 2 variant,

$$\Pr(\text{protocol doesn't abort}) \geq (1 - e^{-1}) \approx 0.632. \quad (4.17)$$

The equation 4.15 implies that if the variant 1-out-of-2 probabilistic oblivious transfer is considered with $\epsilon = \frac{1}{2}$, then the likelihood of Alice receiving the expected number of final key bits and the protocol not aborting is lower bounded by,

$$P_c \geq (1 - e^{-1}) \approx 0.632. \quad (4.18)$$

4.3.3 Security of the protocol

Here, we point towards the security issues of our proposal. We have mentioned the detailed proofs of all our results and showed how these results certify device-independent security, data security, and user security for our proposed scheme.

Based on the results in Corollary 2, Theorem 2, Theorem 3 and Theorem 4, we conclude about the DI security of our proposed scheme. All these results guarantee that either the proposal terminates with high probability (as the limit approaches infinity) or the devices involved in our proposal attain the desired values of the parameters \mathcal{C} , β , Ω^0 and Ω^1 . Later on, we move towards deriving upper bounds on the information gained by dishonest Alice and dishonest Bob. In Lemma 1, we show that dishonest Alice cannot guess (on average) more than $(1 - \cos\theta)$ fraction from the entire raw key. Lemma 2 together with corollary 5 shows that dishonest Bob can guess only $\frac{1}{N}$ fraction from the query index set of Alice.

Security in device independent scenario

The proposed scheme undergoes device independent testing in three phases. The first two are in the *source device verification phase* and *Bob's measurement device verification phase*. The third takes place in *Alice's POVM device verification phase*.

In source device verification phase, at first *LocalCHSH game* has been performed by each of Alice and Bob independently (as mentioned in LocalCHSHtest) at their end for some randomly chosen samples. In this phase, both Alice and Bob test individually whether the states provided by the third party are EPR pairs. Bob and Alice choose the samples randomly for which they want to perform LocalCHSHtest and share this information publicly to get the corresponding qubits from the other party and also to identify all the samples for which they perform LocalCHSHtest.

As QPQ is a distrustful scheme, both the parties may not behave honestly in every phase of the protocol. For this reason, here we assume that the party who acts honestly for a particular phase, will take the responsibilities of the referee as well as the two parties in the CHSH game to ensure the random and independent choice of inputs for the devices involved in the LocalCHSHtest at his end. This guarantees that in LocalCHSHtest, the inputs to the devices are random and independent.

The results from the rigidity of CHSH game in [92, Lemma 4.2] lead us to the following conclusion.

Corollary 2 (Verification of shared states). *The LocalCHSHtest of source device certification phase either detects if Alice's and Bob's devices achieve $\mathcal{C} = \cos^2 \frac{\pi}{8}$, meaning they were given EPR pairs (or the unitary equivalent of the actual states) by the third party, or the scheme is likely to abort in the long run.*

In the next phase, Bob verifies his measurement device. Here, Bob is assumed to act honestly in *Bob's measurement device verification phase*, as it's clear from Lemma 2 that a dishonest Bob who wants to guess Alice's query indices more accurately must let Alice know more data bits in a single query, violating assumption 4 that neither party reveals more information to get more information from the other.

Atfirst Bob starts by randomly choosing inputs for his device and measuring the particles. Then Bob and Alice independently pick samples and discuss publicly. Bob generates random input bits for Alice and announces them publicly, so Alice can measure her particles based on the bits. After measurements, they both publicly share inputs and outputs and calculate β as in the OBStest. From this result, one can conclude the following.

Theorem 2 (Bob's measurement device verification). *In OBStest, either Bob's measurement devices achieve the value of the parameter $\beta = \frac{1}{\sqrt{2}|\cos\theta - \sin\theta|}$ (i.e., his devices correctly measure in $\{|0\rangle, |1\rangle\}$ and $\{|0'\rangle, |1'\rangle\}$ basis where $|0'\rangle = (\cos\theta|0\rangle + \sin\theta|1\rangle)$, $|1'\rangle = (\sin\theta|0\rangle - \cos\theta|1\rangle)$), or the protocol terminates with a high likelihood of failure (as the limit approaches infinity).*

A detailed proof of this theorem is provided in Section 4.5 later, using the same method outlined in [65] for certifying non-maximally incompatible observables.

This implies that the LocalCHSHtest certifies the states provided by the third party and OBStest certifies the projective measurement device (for the specific measurement bases used in OBStest) of Bob. As Bob declares a_i values for all the shared instances before OBStest and Alice randomly chooses some of those instances for OBStest, the successful completion of OBStest also implies that for all the remaining instances (i.e., for the instances which are not chosen for OBStest), Alice's state must be either $|0\rangle\langle 0|$ or $|0'\rangle\langle 0'|$ whenever Bob declares $a_i = 0$ and must be either $|1\rangle\langle 1|$ or $|1'\rangle\langle 1'|$ whenever Bob declares $a_i = 1$.

The third DI test is performed in *Alice's POVM device verification phase*. The protocol moves to this phase once both Alice and Bob have passed the first two DI tests. So, Alice and Bob are in this phase implies that both Bob's projective measurement device and their shared states are noiseless. Now, this testing phase basically guarantees the functionality of Alice's POVM device. Note that in this phase, Bob does not need to test his measurement device again. During OBStest, his devices are tested already. However, Alice has to shift to a new measurement device for better conclusiveness. Device independent security demands that Alice's new device should be tested further for certification. In this phase, Alice measures the selected instances with either device $M^0 = M_0^0, M_1^0, M_2^0$ or $M^1 = M_0^1, M_1^1, M_2^1$ based on the declared a_i values. She calculates Ω^0 and Ω^1 from the measurement outcomes and verifies if they equal $\frac{2\sin^2\theta}{(1+\cos\theta)}$. Theorem 3 shows that, for the instances where $a_i = 0$, if Alice observes that $\Omega^0 = \frac{2\sin^2\theta}{(1+\cos\theta)}$ then it guarantees that the measurement devices are the desired POVM $\{D_0^0, D_1^0, D_2^0\}$ i.e., $M^0 = D^0$. Similarly, Theorem 4 shows that, for the instances where $a_i = 1$, if Alice observes that $\Omega^1 = \frac{2\sin^2\theta}{(1+\cos\theta)}$ then it guarantees that the measurement devices are the desired POVM $\{D_0^1, D_1^1, D_2^1\}$ i.e., $M^1 = D^1$.

Theorem 3 (Verification of Alice’s measurement device M_0). *POVMtest either results in a high probability of termination of this proposed scheme (as the limit approaches infinity), or it guarantees that for the instances where Bob declares $a_i = 0$, Alice’s measurement devices attain $\Omega^0 = \frac{2\sin^2\theta}{1+\cos\theta}$, meaning they are of this specified form (up to a local unitary),*

$$M_0^0 = \frac{1}{(1 + \cos \theta)}(|1'\rangle\langle 1'|) \quad (4.19)$$

$$M_1^0 = \frac{1}{(1 + \cos \theta)}(|1\rangle\langle 1|) \quad (4.20)$$

$$M_2^0 = \mathbb{I} - M_0^0 - M_1^0. \quad (4.21)$$

where $|1'\rangle = \sin \theta|0\rangle - \cos \theta|1\rangle$.

Theorem 4 (Verification of Alice’s measurement device M_1). *POVMtest either results in a high probability of termination of this proposed scheme (as the limit approaches infinity), or it guarantees that for the instances where Bob declares $a_i = 1$, Alice’s measurement devices attain $\Omega^1 = \frac{2\sin^2\theta}{1+\cos\theta}$, meaning they are of this specified form (up to a local unitary),*

$$M_0^1 = \frac{1}{(1 + \cos \theta)}(|0'\rangle\langle 0'|) \quad (4.22)$$

$$M_1^1 = \frac{1}{(1 + \cos \theta)}(|0\rangle\langle 0|) \quad (4.23)$$

$$M_2^1 = \mathbb{I} - M_0^1 - M_1^1. \quad (4.24)$$

where $|0'\rangle = \cos \theta|0\rangle + \sin \theta|1\rangle$.

The proofs of these two theorems are deferred to Section 4.6 entitled *Verification of Alice’s POVM elements*. In the proof, we restate the functionality of the POVM devices in the form of a two party game (namely POVMgame), consider a general form for the single qubit three outcome POVM $\{M_0^0, M_1^0, M_2^0\}$ ($\{M_0^1, M_1^1, M_2^1\}$) and show that if the input states are chosen randomly between $|0\rangle\langle 0|$ ($|1\rangle\langle 1|$) and $|0'\rangle\langle 0'|$ ($|1'\rangle\langle 1'|$) and if $\Omega^0 = \frac{2\sin^2\theta}{1+\cos\theta}$ ($\Omega^1 = \frac{2\sin^2\theta}{1+\cos\theta}$) then $M_0^0 = D_0^0$ ($M_0^1 = D_0^1$), $M_1^0 = D_1^0$ ($M_1^1 = D_1^1$), $M_2^0 = D_2^0$ ($M_2^1 = D_2^1$).

Note: Here, we claim that if Alice and Bob successfully pass the LocalCHSHtest, the OBStest and the POVMtest mentioned in our DI proposal, then in the actual QPQ scheme, none of Alice and Bob can retrieve any additional information in the noiseless scenario. Now, suppose that our claim is wrong i.e., Alice and Bob can pass all the tests mentioned in our scheme and later Alice can retrieve more data bits (than what she intends to know) in a single query or Bob can guess Alice’s query indices with a more certain probability (than his intended probability).

We now discuss this issue in the context of a particular form of *non-i.i.d.* attack, where a specific number of states are independently corrupted (more general attacks

are also possible but these are outside the scope of this work). In this context, we will show that if some of the corrupted states are included during the testing phases, then there is some probability of being caught in the asymptotic limit.

At the beginning of our scheme, the untrusted third party shares all the states with Alice and Bob. As in the *source device certification phase*, both the parties choose the states randomly from the shared instances for the local tests at their end, the dishonest party can not guess beforehand the shared instances that the honest party will choose at his end for the local test. According to our assumption, the dishonest party can not manipulate the honest party's device once the protocol starts. So, to successfully pass the LocalCHSHtest at the honest party's end, the shared states must be EPR pairs as specified in our scheme. This implies that the *source device certification phase* certifies all the states provided by the untrusted third party.

We now explain these things more formally. Let us suppose that initially, the untrusted third party colludes with either the dishonest Alice or the dishonest Bob and shares either \mathcal{K}_A corrupted states in favour of Alice (let us denote this type of states as \mathcal{A} -type) or \mathcal{K}_B corrupted states in favour of Bob (let us denote this type of states as \mathcal{B} -type) among \mathcal{K} shared states. So, while choosing randomly for the LocalCHSHtest at honest Bob's end, the probability that a chosen state is of \mathcal{A} -type is $\frac{\mathcal{K}_A}{\mathcal{K}}$. Similarly, for the LocalCHSHtest at honest Alice's end, the probability that a chosen state is of \mathcal{B} -type is $\frac{\mathcal{K}_B}{\mathcal{K}}$. Let us further assume that for the \mathcal{A} -type states, the value of the parameter \mathcal{C} is \mathcal{C}_A (where $\mathcal{C}_A = \mathcal{C} + \epsilon_A$ such that $\epsilon_A > 0$) and for the \mathcal{B} -type states, the value of the parameter \mathcal{C} is \mathcal{C}_B (where $\mathcal{C}_B = \mathcal{C} + \epsilon_B$ such that $\epsilon_B > 0$).

Now, suppose that only Alice is dishonest and the third party supplies \mathcal{K}_A number of corrupted states (in favour of dishonest Alice) along with $(\mathcal{K} - \mathcal{K}_A)$ actual states. Then, in the localCHSHtest at Bob's end, the probability that a chosen state is not of the \mathcal{A} -type is $(1 - \frac{\mathcal{K}_A}{\mathcal{K}})$. One can easily check that this probability is also same for a chosen state in the final QPQ phase. As, dishonest Alice's aim is to gain as much additional data bits as possible in the final QPQ phase, she needs to choose the value of \mathcal{K}_A such that $(\mathcal{K} - \mathcal{K}_A) = c$ where c is exponentially smaller than \mathcal{K} (i.e., she will try to maximize the probability that a state chosen for the final QPQ phase is of the \mathcal{A} type). Then, the probability that Bob will choose none of the corrupted states (i.e., the \mathcal{A} type states) among his chosen $\frac{\gamma_1 \mathcal{K}}{2}$ states for the LocalCHSHtest at his end is,

$$\left(1 - \frac{\mathcal{K}_A}{\mathcal{K}}\right)^{\frac{\gamma_1 \mathcal{K}}{2}} = \left(\frac{c}{\mathcal{K}}\right)^{\frac{\gamma_1 \mathcal{K}}{2}}.$$

which is very small compared to \mathcal{K} . Similarly, whenever Bob is dishonest, the same thing can be shown for the LocalCHSHtest at honest Alice's end. This implies that if the third party colludes with the dishonest party and supplies corrupted states then the probability that none of those corrupted states are chosen for the localCHSHtest at the honest party's end is very small.

In our scheme, we consider the ideal scenario where there are no channel noise.

So for dishonest Alice, to successfully pass the LocalCHSHtest at the honest Bob's end, the following relation must hold in the noiseless condition.

$$\begin{aligned}\frac{\mathcal{K}_A \mathcal{C}_A}{\mathcal{K}} + \frac{(\mathcal{K} - \mathcal{K}_A) \mathcal{C}}{\mathcal{K}} &= \mathcal{C} \\ \mathcal{K}_A \mathcal{C}_A + (\mathcal{K} - \mathcal{K}_A) \mathcal{C} &= \mathcal{K} \mathcal{C} \\ \mathcal{K}_A (\mathcal{C}_A - \mathcal{C}) &= 0.\end{aligned}$$

Now, replacing the values of \mathcal{C}_A from the relation $\mathcal{C}_A = \mathcal{C} + \epsilon_A$, one can get,

$$\mathcal{K}_A \epsilon_A = 0. \tag{4.25}$$

As the value of $\epsilon_A > 0$, from this relation, one can easily conclude that in the noiseless scenario, the value of \mathcal{K}_A must be zero to successfully pass the LocalCHSHtest at the honest Bob's end. Similarly, one can show that whenever Bob is dishonest, the value of \mathcal{K}_B must be zero to successfully pass the LocalCHSHtest at the honest Alice's end. In practice, for finite number of samples, one can show that the values of \mathcal{K}_A and \mathcal{K}_B must be very small to successfully pass the local test at the honest party's end.

Here, all the states are shared between the two parties before the start of the protocol and the dishonest party can not manipulate the honest party's device after the start of the protocol. In this study, as the focus is on the *i.i.d.* scenario, it is easy to conclude that the protocol will either abort with high probability in the long run, or the LocalCHSHtest will verify that the states shared in the QPQ scheme have the desired value of \mathcal{C} .

The next DI testing is done in *Bob's measurement device verification phase* where Bob and Alice perform distributed test to certify Bob's device. Here, one may think that if Bob is dishonest, then for the instances chosen in *Bob's measurement device verification phase* and in *Alice's POVM device verification phase*, he will measure in the actual measurement basis at his end to detect the fraudulent behaviour of Alice, and later for the instances to be used for the actual QPQ phase, he will measure in some different basis to guess the positions of Alice's known key bits.

The results in [63] already showed that dishonest Bob can't possess both the correct bit values and conclusiveness information of Alice and if he tries to cheat, then it will damage his reputation as a database owner. So, for the QPQ primitive, the server Bob is expected not to cheat. Moreover, the result in Lemma 2 shows that if Bob wants to increase his chances of guessing Alice's query indices with more certainty while following the exact protocol, he must let dishonest Alice know more data bits in a single query, but this goes against assumption 4, which prohibits the parties from leaking more information to gain extra information from the other party. From the discussion in Lemma 1, it is also clear that for our scheme, the client Alice performs optimal strategy at her end. That means dishonest Alice can't retrieve more data bits in a single query without manipulating the shared states and Bob's measurement

device. Thus, to ensure that dishonest Alice will not get any additional data bits, Bob must behave honestly in *Bob's measurement device verification phase* to certify his device after the successful completion of *source device certification phase*.

In our scheme, before the *Bob's measurement device verification phase*, Bob generates a random bit for each of his qubits and measures his qubits accordingly. In the *Bob's measurement device verification phase*, Bob generates random bits for each of the Alice's qubits chosen for *Bob's measurement device verification phase* and declares those bits so that Alice can measure her particles accordingly. As Bob behaves honestly in *Bob's measurement device verification phase* (to restrict Alice from knowing additional data bits) and chooses all the inputs randomly for OBStest, there is no possibility that the inputs for OBStest are chosen according to some dishonest distribution. From the analysis of Theorem 2, it is clear that if the inputs are chosen randomly then OBStest certifies that Bob's measurement device measures correctly in $\{|0\rangle, |1\rangle\}$ and $\{|0'\rangle, |1'\rangle\}$ basis for our proposed QPQ scheme.

This implies that the successful completion of *source device certification phase* and *Bob's measurement device verification phase* certifies that the shared states are EPR pairs and Bob's measurement device measures correctly for all the instances. This also implies that for all the remaining instances (that will be used for *Alice's POVM device verification phase* and in the actual QPQ phase), Alice has non-orthogonal qubits (i.e., either $|0\rangle$ or $|0'\rangle$ for $a_i = 0$ and either $|1\rangle$ or $|1'\rangle$ for $a_i = 1$) at her end.

It is already mentioned that in our scheme, the client Alice performs optimal (POVM) measurement at her end to extract maximal number of data bits conclusively in a single query. So, after successful completion of *source device certification phase* and *Bob's measurement device verification phase*, Alice must behave honestly in *Alice's POVM device verification phase* to ensure that her measurement device is the optimal one. For this reason, Alice must measure her qubits accordingly as mentioned in KEYgen() and POVMtest() to certify her device. From the analysis of Theorem 3 and Theorem 4, it is clear that the successful completion of *Alice's POVM device verification phase* certifies Alice's POVM device.

Note that in the proof of Theorem 3 and Theorem 4 in Section 4.6 (entitled *Verification of Alice's POVM Elements*), we have not imposed any dimension bound like the self-testing of POVM in a prepare and measure scenario in [106]. So, the devices that perform a Neumark dilation of this mentioned POVM (i.e., the equivalent larger projective measurement on both the original state and some ancilla system instead of the actual POVM measurement) could still achieve the intended value of Ω . But both of these operations produce the same output probabilities, which is sufficient for the purposes of this work.

Therefore, from all these discussions, one can conclude the following.

Corollary 3. *Our DI proposal either terminates with high likelihood (as the limit approaches infinity), or it confirms that the devices in our QPQ proposal meet the target values of \mathcal{C} , β , and Ω^0 (or Ω^1) in the LocalCHSHtest, OBStest, and POVMtest respectively.*

Security of database against dishonest Alice

In this subsection, we calculate the number of raw key bits that an dishonest Alice can determine in our proposed scheme's *shared key generation phase*.

Theorem 5. *In our proposal, in the absence of POVMtest, dishonest Alice can retrieve, at most, $(\frac{1}{2} + \frac{1}{2} \sin \theta)$ fraction of bits from the entire raw key, inconclusively (i.e., the indices of the correctly guessed bits are unknown), during the shared key generation phase.*

Proof. At the end of the *shared key generation phase*, dishonest Alice (\mathcal{A}^*) and honest Bob (\mathcal{B}) share kN raw key bits obtained from kN EPR pairs. The i -th copy of the state is given by $|\phi^+\rangle_{\mathcal{A}_i^* \mathcal{B}_i} = \frac{1}{\sqrt{2}}|00\rangle_{\mathcal{A}_i^* \mathcal{B}_i} + \frac{1}{\sqrt{2}}|11\rangle_{\mathcal{A}_i^* \mathcal{B}_i}$, where i -th subsystem of Alice and Bob is denoted by \mathcal{A}_i^* and \mathcal{B}_i respectively. At Alice's side the reduced density matrix is of the form

$$\rho_{\mathcal{A}_i^*} = \text{Tr}_{\mathcal{B}_i} [|\phi^+\rangle_{\mathcal{A}_i^* \mathcal{B}_i} \langle \phi^+|] = \frac{\mathbb{I}_2}{2}.$$

At the beginning, Bob measures each of his part of the state $|\phi^+\rangle_{\mathcal{A}_i^* \mathcal{B}_i}$ in either $\{|0\rangle, |1\rangle\}$ basis or in $\{|0'\rangle, |1'\rangle\}$ basis. The choice of the basis is completely random as this choice depends on the random raw key bit values chosen by Bob. Let $\rho_{\mathcal{A}_i^* | r_i}$ denotes the state at Alice's side after the choice of Bob's measurement basis. For $r_i = 0$, we have,

$$\begin{aligned} \rho_{\mathcal{A}_i^* | r_i=0} &= \text{Tr}_{\mathcal{B}_i} [|\phi^+\rangle_{\mathcal{A}_i^* \mathcal{B}_i} \langle \phi^+|] \\ &= \text{Tr}_{\mathcal{B}_i} \left[\frac{1}{2} (|00\rangle + |11\rangle)_{\mathcal{A}_i^* \mathcal{B}_i} (\langle 00| + \langle 11|) \right] \\ &= \frac{\mathbb{I}_2}{2}. \end{aligned}$$

Similarly, for $r_i = 1$, we have, $\rho_{\mathcal{A}_i^* | r_i=1} = \frac{\mathbb{I}_2}{2} = \rho_{\mathcal{A}_i^*}$. This implies that $\rho_{\mathcal{A}_i^* | r_i} = \rho_{\mathcal{A}_i^*}$. In *Bob's measurement device verification phase*, Alice knows the declared a_i values for all the instances. Let $\rho_{\mathcal{A}_i^* | a_i}$ denotes the state of Alice given the value of a_i . According to the protocol,

$$\begin{aligned} \rho_{\mathcal{A}_i^* | a_i=0} &= \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|0'\rangle\langle 0'| \\ \rho_{\mathcal{A}_i^* | a_i=1} &= \frac{1}{2}|1\rangle\langle 1| + \frac{1}{2}|1'\rangle\langle 1'|. \end{aligned}$$

This implies that for a fixed $a_i = 0$ ($a_i = 1$) if Alice wants to guess the value of r_i then she needs to distinguish the state from the ensemble of states $\{(\frac{1}{2}|0\rangle\langle 0|), (\frac{1}{2}|0'\rangle\langle 0'|)\}$ ($\{(\frac{1}{2}|1\rangle\langle 1|), (\frac{1}{2}|1'\rangle\langle 1'|)\}$). In other words, whenever Bob measures his qubit and announces the bit $a_i = 0$, Alice knows that Bob gets either $|0\rangle$ or $|0'\rangle$. Similarly, when Bob announces the bit $a_i = 1$, Alice knows that Bob gets either $|1\rangle$ or $|1'\rangle$. So, to retrieve the value of the original raw key bit, Alice needs to distinguish between the states $|0\rangle$ and $|0'\rangle$ or between the states $|1\rangle$ or $|1'\rangle$.

Now, in the absence of the POVMtest (i.e., if Alice’s measurement device is not tested), Alice can choose any measurement device at her side to distinguish the non-orthogonal states generated at her side. As it is known that non-orthogonal quantum states cannot be distinguished perfectly, Alice cannot guess the value of each raw key bit with certainty. This distinguishing probability has a nice relationship with the trace distance between the states in the ensemble [58]. According to this relation we have,

$$\begin{aligned}\Pr_{\text{guess}} [r_i | \rho_{\mathcal{A}_i^* | a_i=0}] &= \frac{1}{2} \left(1 + \frac{1}{2} \|\!|0\rangle\langle 0| - |0'\rangle\langle 0'|\!|\!|_1\right) \\ &\leq \frac{1}{2} \left(1 + \sqrt{1 - F(|0\rangle\langle 0|, |0'\rangle\langle 0'|)}\right) \\ &= \frac{1}{2} (1 + \sin \theta) = \frac{1}{2} + \frac{1}{2} \sin \theta.\end{aligned}$$

One can check that $\Pr_{\text{guess}} [r_i | \rho_{\mathcal{A}_i^* | a_i=0}] = \Pr_{\text{guess}} [r_i | \rho_{\mathcal{A}_i^* | a_i=1}]$. This implies that if Alice is allowed to use any measurement device at her end after *Bob’s measurement device verification phase* then Alice can successfully retrieve the i -th raw key bit r_i with probability at most $(\frac{1}{2} + \frac{1}{2} \sin \theta)$. As after *Bob’s measurement device verification phase*, the qubits at Alice’s side are all independent, dishonest Alice can inconclusively retrieve (on average) atmost $(\frac{1}{2} + \frac{1}{2} \sin \theta)$ fraction of bits of the entire raw key. \square

Note : Here, the term ‘inconclusive’ means that Alice can’t determine the positions of the accurately guessed key bits with certainty. For example, whenever Alice tries to guess each of the key bits randomly, she can guess correctly for around half of the instances. However, she can’t tell with certainty what are those instances for which she guesses correctly.

Now let us consider the operator $E = \{E_0, E_1, E_2\}$ where,

$$\begin{aligned}E_0 &\equiv \frac{1}{\sin \theta} (\sin \theta |0\rangle - \cos \theta |1\rangle)(\sin \theta \langle 0| - \cos \theta \langle 1|) \\ E_1 &\equiv \frac{1}{\sin \theta} |1\rangle \langle 1| \\ E_2 &\equiv I - E_0 - E_1\end{aligned}$$

One can easily check that this operator $E = \{E_0, E_1, E_2\}$ is not a valid POVM as E_2 is not positive semi-definite. Let us consider the operator $E' = \{E'_0, E'_1\}$ where

$$\begin{aligned}E'_0 &\equiv E_0 + \frac{E_2}{2} \\ E'_1 &\equiv E_1 + \frac{E_2}{2}\end{aligned}$$

Now, this is a valid POVM to distinguish $|0\rangle$ and $|0'\rangle = (\cos \theta |0\rangle + \sin \theta |1\rangle)$. If a party considers the strategy that for the outcome E'_0 , he considers the corresponding input qubit as $|0\rangle$ and $|0'\rangle$ otherwise, then one can check that this is the POVM

corresponding to the optimal success probability (i.e., $\frac{1}{2} + \frac{\sin \theta}{2}$) in distinguishing $|0\rangle$ and $|0'\rangle$. However, the guessing outcome of this POVM is uncertain as the inconclusive element (the outcome which can't determine the state with certainty) E_2 is involved in both the elements E'_0 and E'_1 of the POVM E' . So, in the proof of Theorem 5, we refer the optimal guessing probability as inconclusive (i.e., uncertainty about the positions of the known key bits).

In theorem 5, we show that if Alice is allowed to choose any measurement device at her side then, on average, dishonest Alice can correctly retrieve at most around $(\frac{1}{2} + \frac{\sin \theta}{2})$ fraction from the entire raw key but she remains uncertain about the positions of those known bits.

However, in this DI proposal, dishonest Alice's (\mathcal{A}^*) main intention is to conclusively (i.e., with certainty about the positions of the correctly guessed key bits) retrieve as many raw key (as well as final key) bits as possible because otherwise she can't know which data bits she has retrieved correctly. For this reason, dishonest Alice has to perform the mentioned POVM measurement at her end to retrieve maximum number of raw key bits conclusively. Because of this, one can get a bound on the number of raw key bits that dishonest Alice can guess (on average) in this DI-QPQ proposal.

Lemma 1. *Either this proposed DI-QPQ scheme terminates with high likelihood in the long run, or dishonest Alice (\mathcal{A}^*) can retrieve (on average) $(1 - \cos \theta)$ fraction of bits from the entire raw key after the shared key generation phase of our proposal.*

Proof. According to our proposal, after the Alice's POVM device verification phase, the client Alice has kN independent non-orthogonal qubits at her end. For each of these instances, dishonest Alice now tries to distinguish between the non-orthogonal states either $|0\rangle$ and $|0'\rangle$ (for $a_i = 0$) or $|1\rangle$ and $|1'\rangle$ (for $a_i = 1$).

In this regard, she chooses the measurement device $\{M_0^0, M_1^0, M_2^0\}$ when Bob announces $a_i = 0$ and measurement device $\{M_0^1, M_1^1, M_2^1\}$ when Bob announces $a_i = 1$.

Whenever the outcome is M_0^0 (M_0^1), Alice concludes that the state is $|0\rangle$ ($|1\rangle$). If it is M_1^0 (M_1^1), she concludes that the state is $|0'\rangle$ ($|1'\rangle$). The guessing remains inconclusive (i.e., can't guess the outcome with certainty) only when the measurement outcome is M_2^0 (M_2^1).

It is evident from [62] that the maximum probability of successfully distinguishing two non-orthogonal states is $(1 - \cos \theta)$. From Theorem 1, we get that in our protocol, the success probability of Alice in guessing a key bit correctly and conclusively is also $(1 - \cos \theta)$. As Alice has to measure each of her qubits independently depending on the declared a_i values, on average she can conclusively retrieve $(1 - \cos \theta)$ fraction from the entire raw key. This concludes the proof. □

Dishonest Alice can employ a broader attack strategy by storing all the photons in a quantum memory and deferring the measurements until the initial step of the *private query phase* where Bob discloses the qubits that contribute to each final key bit. In this scenario, without performing optimal measurements individually for each

of the k qubits, dishonest Alice can perform a joint optimal measurement on all the k qubits associated with a final key bit to extract the key bits. It is well-known that the probability of correctly identifying one of two equally likely quantum states (say ρ_0 and ρ_1) is upper bounded by $\frac{1}{2} + \frac{1}{2}D(\rho_0, \rho_1)$, where $D(\rho_0, \rho_1)$ represents the trace distance. In the case of a joint Helstrom measurement by dishonest Alice on k qubits (related to a final key bit in our proposal), this probability boils down to $\left(\frac{1}{2} + \frac{\sin^k \theta}{2}\right)$ as the number of added qubits (k) increases. Furthermore, as explained in the note following the proof of Theorem 5, this optimal measurement would be inconclusive. In other words, dishonest Alice cannot accurately determine the indices of her known final key bits, which is a crucial requirement for the QPQ primitive. Therefore, this joint measurement attack is ineffective for our (and any other) QPQ proposal.

Although there is a chance that dishonest Alice can successfully pass all tests and learn more data bits than allowed through statistical fluctuations, the likelihood of this happening is low according to Corollary 3. Now from Definition 3 and equation 4.6, we can conclude the following.

Corollary 4. *In the case of dishonest Alice and honest Bob, either our proposal will terminate (as the limit approaches infinity) or dishonest Alice will, on average, be able to retrieve τ fraction of bits from the entire final key, where*

$$\tau \leq (1 - \cos \theta)^k. \quad (4.26)$$

By using the upper bound from equation 4.7 in place of $(1 - \cos \theta)^k$, one can obtain the following bound on the value of τ .

$$\boxed{\tau < \frac{2}{N}}. \quad (4.27)$$

This relation signifies that in our DI-QPQ proposal, τ is significantly smaller than N .

Now, we validate the probabilistic definition of data privacy for this proposed scheme and show that the probability $\Pr[|X - Y| > \delta \wedge \text{scheme doesn't terminate}]$ is negligible (where, similar to our previous definition, X and Y denote the actual and expected number of final key bits respectively for Alice). More specifically, we will calculate the probability with which dishonest Alice can guess more than the expected number of final key bits (with a deviation more than the ϵ fraction of the expected number of final key bits).

The negligibility of the probability $\Pr[|X - Y| > \delta \wedge \text{scheme doesn't terminate}]$ can be shown using the properties of basic probability theory. Note that the probability $\Pr[|X - Y| > \delta \wedge \text{scheme doesn't terminate}]$ is upper bounded by both $\Pr[|X - Y| > \delta]$ and $\Pr[\text{scheme doesn't terminate}]$, according to the properties $\Pr[A \wedge B] \leq \Pr[A]$ and $\Pr[A \wedge B] \leq \Pr[B]$. As in our scheme, we consider the *i.i.d.* assumption, there will be two different subcases- 1) all the devices attain ideal values in all the testing phases (i.e., in LocalCHSHtest, OBStest and POVMtest) 2) all the devices don't attain ideal values in all the testing phases.

For the first subcase, from the correctness result (i.e., the value of P_c for our scheme in equation 4.15) and the DI security statement in Corollary 3, one can easily conclude that $\Pr[|X - Y| > \delta] \leq \text{negl}(N)$ where $\text{negl}(N)$ denotes negligible in N . For the second subcase, by an analysis similar to the proof of Theorem 2 and from the DI security statement in Corollary 3, it can be concluded that $\Pr[\text{scheme doesn't terminate}] \leq \text{negl}(N)$. This implies that for both of these two subcases, $\Pr[|X - Y| > \delta \wedge \text{scheme doesn't terminate}] \leq \text{negl}(N)$ (under the *i.i.d.* assumption).

Although it is easy to derive the negligibility of the expression $\Pr[|X - Y| > \delta \wedge \text{scheme doesn't terminate}]$ for both the two subcases, in general for the second subcase, it is hard to derive the exact bound on the probability with which dishonest Alice can guess more than the expected number of final key bits. For our proposed scheme, as Alice performs optimal POVM measurement at her end, it is relatively easier to derive an upper bound on the parameter P_d for our scheme because it is unlikely that dishonest Alice can retrieve more number of raw key bits (on average) by performing any other measurements at her end.

To derive the exact bound on the parameter P_d for this proposal, like the previous discussion considering X and Y be the actual and expected number of final key bits respectively for Alice, here from the Chernoff-Hoeffding inequality [59] mentioned in proposition 1, one can conclude the following.

$$\begin{aligned} & \Pr[|X - Y| \geq \delta_{CH} \wedge \text{scheme doesn't terminate}] \\ & \leq \exp(-2\delta_{CH}^2 N). \end{aligned} \tag{4.28}$$

Here, we aim to calculate the likelihood of the value of X being outside the error range of $\delta_{CH} = \epsilon(1 - \cos \theta)^k N$ from its expected value.

From the relation in equation 4.7, it can be easily derived that whenever Bob selects θ so that $(1 - \cos \theta)^k = \frac{1}{N}$, the equation 4.28 becomes,

$$\begin{aligned} & \Pr[|X - Y| \geq \epsilon \wedge \text{scheme doesn't terminate}] \\ & \leq \exp(-2\epsilon^2 N). \end{aligned} \tag{4.29}$$

So, from definition 3, the parameter P_d in our proposed scheme (that corresponds to dishonest Alice and honest Bob) can be upper bounded by,

$$\boxed{P_d \leq \exp(-2\epsilon^2 N)}. \tag{4.30}$$

That means the probability that dishonest Alice can learn more than the expected amount of final key bits (beyond the ϵ deviation) while the protocol doesn't abort is very low in practice because the value of N is very large.

For the purpose of illustration, again we evaluate here the performance of our scheme as a 1 out of 2 probabilistic oblivious transfer, where $N = 2$ and $k = 1$. From expression 4.30, with an error margin of $\epsilon = \frac{1}{2}$, the probability that dishonest

Alice can guess more than the expected number of final key bits (which is 1) is upper bounded by,

$$P_d \leq e^{-1} \approx 0.368. \quad (4.31)$$

The comparison between the highest probability of inconclusive success (i.e., uncertainty in guessing the position of correct bits) and the highest probability of conclusive success (i.e., ability to accurately guess the position of correct bits) is depicted in Figure 4-2. The figure demonstrates that for small values of θ , the highest inconclusive success probability surpasses the highest conclusive success probability.

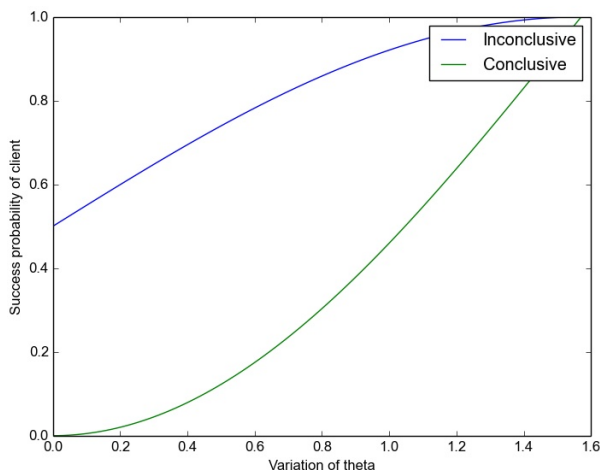


Figure 4-2: Comparison between maximum inconclusive and conclusive success probability of the client.

Security of Alice against dishonest Bob

In this subsection, we determine the number of indices (l_{B^*}) that dishonest Bob can correctly guess from \mathcal{I}_l (the query index set of Alice) and calculate the probability of Bob correctly guessing more than the expected number of indices. In [63], it was discussed that the dishonest Bob could employ a middle-state attack to gain insight into Alice’s conclusiveness or her known bit values. However, it is impossible (shown in [63]) for dishonest Bob to possess knowledge of both the correct bit values and the conclusiveness information. Engaging in systematic cheating would result in incorrect answers provided to Alice, damaging Bob’s reputation as a database provider. Thus, in the QPQ primitive, Bob is expected to adhere to the actual protocol, as there exists a non-zero probability of being caught cheating.

Lemma 2. *Dishonest Bob can correctly guess a maximum of $\frac{l}{N}$ fraction of the indices from the query set \mathcal{I}_l of Alice after l queries to the N -bit database, i.e., for a particular*

index i ,

$$\Pr(\text{Bob correctly guesses } i \in \mathcal{I}_l) \leq \frac{l}{N}.$$

Proof. In the *shared key generation phase* of our proposal, Alice does not broadcast anything about her measurement outcome. So, dishonest Bob has no information about Alice's measurement outcomes and her known key bits. Now, Alice queries l many times to the database and retrieves l many data bits. After these l many queries, dishonest Bob will try to guess those query indices of Alice. As Bob has no information about Alice's known final key bits, he has no other options other than randomly guessing these l many indices (out of the N data bits).

So, for any i -th data bit, dishonest Bob can guess whether $i \in \mathcal{I}_l$ with probability atmost $\frac{l}{N}$. This completes the proof. \square

This means that when Bob makes a guess at a specific data bit index, the chance of it being in Alice's query index set is roughly $\frac{l}{N}$. Assuming that after l queries, the set \mathcal{I}_l of Alice's query indices has l data bits, and the chosen indices are independent, the expected number of indices ($l_{\mathcal{B}^*}$) that dishonest Bob correctly guesses from \mathcal{I}_l would be,

$$\begin{aligned} E[l_{\mathcal{B}^*}] &= \Pr(\text{Bob correctly guesses } i \in \mathcal{I}_l).l \\ &\leq \frac{l^2}{N}. \end{aligned} \tag{4.32}$$

However, this guess will be inconclusive i.e., Bob can't identify his correctly guessed indices with certainty because of the random guess. Now, comparing the expression in definition 4 with equation 4.32 provides the following upper bound for δ in our proposal.

Corollary 5. *Our DI-QPQ proposal either terminates with high likelihood in the long run, or dishonest Bob can guess, on average, δ fraction of indices in Alice's query index set \mathcal{I}_l where,*

$$\delta \leq \left(\frac{l}{N} \right). \tag{4.33}$$

The typical size of the database is much larger (exponentially so) than the size of Alice's query index set, i.e., $N = l^n$, where n is a positive integer ($n > 1$). Plugging this into equation 4.33 gives the following upper bound on the value of δ .

$$\boxed{\delta \leq \frac{1}{l^{(n-1)}}}. \tag{4.34}$$

This relation shows that for our DI-QPQ proposal, δ is significantly smaller compared to l .

Now, we validate the probabilistic definition of user privacy against dishonest Bob for our full DI proposal and derive the exact bound on the security parameter P_u . As shown in Lemma 1, dishonest Bob's chance of guessing if an index i is in the index

set \mathcal{I}_l (of Alice) is limited to $\frac{l}{N}$. Also, this upper bound is determined incorporating the scenario that the proposal doesn't terminate. This implies that,

$$\begin{aligned} & \Pr [\text{Bob guesses } i \in \mathcal{I}_l \wedge \text{scheme doesn't terminate}] \\ & \leq \frac{l}{N}. \end{aligned} \tag{4.35}$$

So, from definition 4, the parameter P_u in our proposed scheme (that corresponds to dishonest Bob and honest Alice) can be upper bounded by,

$$\boxed{P_u \leq \frac{l}{N}}. \tag{4.36}$$

In practice, the probability of dishonest Bob correctly guessing a database index in Alice's query index set is low due to a large difference in size between the database (N) and query index set (l).

Here also, we evaluate the performance of our proposal considering it as 1-out-of-2 probabilistic oblivious transfer (i.e., $N = 2$, $k = 1$ and $l = 1$). From expression 4.36, we get that the value of P_u for our scheme is upper bounded by,

$$P_u \leq \frac{1}{2} \approx 0.5. \tag{4.37}$$

4.4 Choice of initial sample size in practice

In this section, we discuss how Bob and Alice choose the initial sample size required for the proposed DI-QPQ scheme. In practice, Alice and Bob have to allow some deviation (from the actual value of the parameter because of finite number of samples) in each testing phase to certify the devices.

It is well-known that the approximate number of samples required to distinguish two events having probabilities p and $p(1 + \epsilon)$ (for small ϵ) is $O(\frac{1}{p\epsilon^2})$. One may require approximately $\frac{64}{p\epsilon^2}$ samples to achieve a confidence of more than 99% in distinguishing these two events. A more involved expression of the sample size is recently derived in [16] using Chernoff-Hoeffding [59] bound which is stated in proposition 1.

For the testing phases mentioned in our proposed scheme, we consider $X_i = 1$ whenever Bob and Alice win the i -th instance and $X_i = 0$ otherwise. Now if we consider $\mathbb{E}[X_i] = p$ and want to estimate the success probability p within an error margin of ϵp and confidence $1 - \eta$, then from the result mentioned in [16], we can write that the required sample size m_{req} will be,

$$m_{\text{req}} \geq \frac{1}{2\epsilon^2 p^2} \ln \frac{1}{\eta}. \tag{4.38}$$

From this expression of m_{req} , Bob and Alice can estimate the expected number of samples required for a particular testing phase to certify a device with certain accuracy and confidence.

Now to ensure that Bob and Alice get the expected number of samples in each phase (to conclude with certain accuracy and confidence), they choose the total initial sample size (i.e., the value of \mathcal{K}) as follows-

- Before the start of the protocol, Alice and Bob (based on the protocol description) calculate the minimum number of samples required (according to the expression in inequality 4.38) in each testing phase to conclude with chosen accuracy and confidence.
- Then they choose the value of k to calculate the total number of samples required in *private query phase*.
- At last, they sum up all these number of samples required in each testing phase along with the number of samples required in private query phase to calculate the total initial sample size.
- After getting the initial sample size, Bob and Alice proceed to each of the testing phases (according to the description of the protocol), select the required number of samples randomly from the shared instances and check whether the value of a predefined parameter lies within the interval $[V - \epsilon p, V + \epsilon p]$ where V is the actual value of the parameter obtained for asymptotically large number of samples. If this is the case, then with accuracy ϵp and chosen confidence $(1 - \eta)$, they conclude that the devices behave accordingly.

As an example, here we demonstrate the method of choosing samples for the first phase namely source device verification phase. Before the start of the protocol, Bob and Alice choose the accuracy and confidence parameter for this phase with which they want to certify the source device and let n_1 be the required number of samples. Now, similar to this *source device certification phase*, they calculate the required number of samples for the other phases also and from that calculate the required number of total initial samples \mathcal{K} .

Bob and Alice then calculate the value of γ_1 such that,

$$n_1 = \gamma_1 \mathcal{K}.$$

After getting the value of γ_1 , Bob first chooses $\frac{\gamma_1 \mathcal{K}}{2}$ number of samples randomly from the \mathcal{K} shared states and then from the rest $(\mathcal{K} - \frac{\gamma_1 \mathcal{K}}{2})$ number of samples, Alice randomly chooses $\frac{\gamma_1 \mathcal{K}}{2}$ number of samples. They then discuss their chosen instances publicly, get the qubits from the other party and perform LocalCHSHtest for their chosen $\gamma_1 \mathcal{K}$ samples. In this similar way, they choose the samples for the remaining testing phases.

Note that this is a particular way of choosing samples that we demonstrate here from the several other possibilities. It is needless to say that one may follow any other strategies for choosing samples in different testing phases.

4.5 Statement and proof of Theorem 2

Theorem 2: In OBStest, either Bob's measurement devices achieve the value of the parameter $\beta = \frac{1}{\sqrt{2}|\cos\theta - \sin\theta|}$ (i.e., his devices correctly measure in $\{|0\rangle, |1\rangle\}$ and $\{|0'\rangle, |1'\rangle\}$ basis where $|0'\rangle = (\cos\theta|0\rangle + \sin\theta|1\rangle)$, $|1'\rangle = (\sin\theta|0\rangle - \cos\theta|1\rangle)$), or the protocol terminates with a high likelihood of failure (as the limit approaches infinity).

Proof: Suppose, Alice's measurement operators are $\{A_c^s\}_{s,c \in \{0,1\}}$, corresponding to the input s and output c . Similarly, Bob's measurement operators are $\{B_b^r\}_{r,b \in \{0,1\}}$, corresponding to the input r and output b . This implies that Alice's observable, corresponding to the input $s \in \{0, 1\}$ is,

$$A_s = \sum_{c \in \{0,1\}} (-1)^c A_c^s. \quad (4.39)$$

Similarly, Bob's observable corresponding to the input $r \in \{0, 1\}$ is,

$$B_r = \sum_{b \in \{0,1\}} (-1)^b B_b^r. \quad (4.40)$$

Note that, in the OBStest, the fraction β is being computed as follows,

$$\beta = \frac{1}{4} \sum_{s,r,c,b \in \{0,1\}} (-1)^{d_{srcb}} \alpha^{1 \oplus s} \langle \phi_{AB} | A_c^s \otimes B_b^r | \phi_{AB} \rangle \quad (4.41)$$

$$= \frac{1}{4} \langle \phi_{AB} | W_{\mathcal{A}} | \phi_{AB} \rangle. \quad (4.42)$$

where $W_{\mathcal{A}} := \left(\sum_{s,r,c,b \in \{0,1\}} (-1)^{d_{srcb}} \alpha^{1 \oplus s} A_c^s \otimes B_b^r \right)$ which is the operator corresponding to the OBStest. We can also rewrite the expression of $W_{\mathcal{A}}$ in the following way.

$$\begin{aligned} W_{\mathcal{A}} &= \left(\sum_{r,c,b \in \{0,1\}} (-1)^{d_{srcb}} \alpha A_c^0 \otimes B_b^r \right) + \\ &\quad \left(\sum_{r,c,b \in \{0,1\}} (-1)^{d_{srcb}} A_c^1 \otimes B_b^r \right) \\ &= W_{\mathcal{A}}^0 + W_{\mathcal{A}}^1. \end{aligned} \quad (4.43)$$

where $W_{\mathcal{A}}^0 := \left(\sum_{r,c,b \in \{0,1\}} (-1)^{d_{srcb}} \alpha A_c^0 \otimes B_b^r \right)$ and $W_{\mathcal{A}}^1 := \left(\sum_{r,c,b \in \{0,1\}} (-1)^{d_{srcb}} A_c^1 \otimes B_b^r \right)$. Note that, we can simplify further the expression of $W_{\mathcal{A}}^0$ in following way.

$$\begin{aligned}
W_{\mathcal{A}}^0 &= \sum_{r,c,b \in \{0,1\}} (-1)^{d_{srcb}} \alpha A_c^0 \otimes B_b^r \\
&= \sum_{\substack{r,c,b \in \{0,1\} \\ c \oplus b = 0}} \alpha A_c^0 \otimes B_b^r - \sum_{\substack{r,c,b \in \{0,1\} \\ c \oplus b \neq 0}} \alpha A_c^0 \otimes B_b^r \\
&= \alpha(A_0^0 \otimes B_0^0 + A_0^0 \otimes B_0^1 + A_1^0 \otimes B_1^0 + A_1^0 \otimes B_1^1) - \\
&\quad \alpha(A_0^0 \otimes B_1^0 + A_0^0 \otimes B_1^1 + A_1^0 \otimes B_0^0 + A_1^0 \otimes B_0^1) \\
&= \alpha[A_0^0 \otimes (B_0^0 - B_1^0) - A_1^0 \otimes (B_0^0 - B_1^0) + \\
&\quad A_0^0 \otimes (B_0^1 - B_1^1) - A_1^0 \otimes (B_0^1 - B_1^1)] \\
&= \alpha[(A_0^0 - A_1^0) \otimes (B_0^0 - B_1^0) + (A_0^0 - A_1^0) \otimes (B_0^1 - B_1^1)] \\
&= \alpha(A_0^0 - A_1^0) \otimes [(B_0^0 - B_1^0) + (B_0^1 - B_1^1)].
\end{aligned}$$

By substituting the values of $(A_0^0 - A_1^0)$, $(B_0^0 - B_1^0)$ and $(B_0^1 - B_1^1)$ from equation 4.39 and equation 4.40 on the right-hand side of the above expression we get,

$$W_{\mathcal{A}}^0 = \alpha A_0 \otimes (B_0 + B_1). \quad (4.44)$$

Using similar approach we get the following simplified version of the expression $W_{\mathcal{A}}^1$.

$$W_{\mathcal{A}}^1 = A_1 \otimes (B_0 - B_1). \quad (4.45)$$

By substituting the values of $W_{\mathcal{A}}^0$ and $W_{\mathcal{A}}^1$ from equation 4.44 and equation 4.45 to equation 4.43 we get,

$$W_{\mathcal{A}} = \alpha A_0 \otimes (B_0 + B_1) + A_1 \otimes (B_0 - B_1). \quad (4.46)$$

Note that, the right-hand side of this OBStest operator $W_{\mathcal{A}}$ is exactly same as the tilted CHSH operator, described in [65].

So, the expression of $W_{\mathcal{A}}^2$ can be written as

$$\begin{aligned}
W_{\mathcal{A}}^2 &= \alpha^2 A_0^2 \otimes (B_0^2 + B_1^2 + \{B_0, B_1\}) \\
&\quad + A_1^2 \otimes (B_0^2 + B_1^2 - \{B_0, B_1\}) \\
&= (\alpha^2 A_0^2 + A_1^2 + \alpha\{A_0, A_1\}) \otimes B_0^2 \\
&\quad + (\alpha^2 A_0^2 + A_1^2 - \alpha\{A_0, A_1\}) \otimes B_1^2 \\
&\quad + (\alpha^2 A_0^2 - A_1^2) \otimes \{B_0, B_1\} - \alpha[A_0, A_1] \otimes [B_0, B_1].
\end{aligned}$$

Using the property $A_j^2 \leq \mathbb{I}$, we can rewrite this expression as,

$$\begin{aligned}
W_{\mathcal{A}}^2 &\leq [(\alpha^2 + 1).\mathbb{I} + \alpha\{A_0, A_1\}] \otimes B_0^2 \\
&\quad + [(\alpha^2 + 1).\mathbb{I} - \alpha\{A_0, A_1\}] \otimes B_1^2 \\
&\quad + \mathbb{I} \otimes (\alpha^2 - 1)\{B_0, B_1\} - \alpha[A_0, A_1] \otimes [B_0, B_1].
\end{aligned}$$

Since $-2.\mathbb{I} \leq \{A_0, A_1\} \leq 2.\mathbb{I}$, we have,

$$[(\alpha^2 + 1).\mathbb{I} \pm \alpha\{A_0, A_1\}] \geq 0.$$

We can use the property $B_k^2 \leq \mathbb{I}$ and get the following simplified expression.

$$\begin{aligned}
W_{\mathcal{A}}^2 &\leq 2(\alpha^2 + 1).\mathbb{I} \otimes \mathbb{I} + \mathbb{I} \otimes (\alpha^2 - 1)\{B_0, B_1\} \\
&\quad - \alpha[A_0, A_1] \otimes [B_0, B_1].
\end{aligned}$$

We can further upper bound the commutators by their matrix modulus and use the relation $|[A_0, A_1]| \leq 2.\mathbb{I}$ to get the following expression.

$$W_{\mathcal{A}}^2 \leq 2(\alpha^2 + 1).\mathbb{I} \otimes \mathbb{I} + T_{\alpha} \otimes \mathbb{I}. \quad (4.47)$$

where $T_{\alpha} := (\alpha^2 - 1)\{B_0, B_1\} + 2\alpha|[B_0, B_1]|$

Now the expression of T_{α} can also be upper bounded by upper bounding the anti commutators by its matrix modulus. So, the value of T_{α} will be upper bounded by,

$$T_{\alpha} \leq (\alpha^2 - 1)|\{B_0, B_1\}| + 2\alpha|[B_0, B_1]|.$$

Again one can easily check that,

$$\begin{aligned}
&|\{B_0, B_1\}|^2 + |[B_0, B_1]|^2 \\
&= |B_0B_1 + B_1B_0|^2 + |B_0B_1 - B_1B_0|^2 \\
&= (B_0B_1 + B_1B_0)^{\dagger}(B_0B_1 + B_1B_0) \\
&\quad + (B_0B_1 + B_1B_0)^{\dagger}(B_0B_1 + B_1B_0) \\
&= 2(B_0B_1)^{\dagger}(B_0B_1) + 2(B_1B_0)^{\dagger}(B_1B_0).
\end{aligned} \quad (4.48)$$

Let us consider that the measurement operators are projective i.e., $(A_c^s)^2 = A_c^s$ and $(B_b^r)^2 = B_b^r$. Now for the projectors B_0^0 and B_1^0 , $(B_0^0 + B_1^0) = \mathbb{I}$. From this relation we can write,

$$\begin{aligned}
(B_0^0 + B_1^0)(B_0^0 + B_1^0)^\dagger &= \mathbb{I} \\
B_0^0 \cdot B_0^{0\dagger} + B_0^0 \cdot B_1^{0\dagger} + B_1^0 \cdot B_0^{0\dagger} + B_1^0 \cdot B_1^{0\dagger} &= \mathbb{I} \\
(B_0^0 + B_1^0) + (B_0^0 \cdot B_1^{0\dagger} + B_1^0 \cdot B_0^{0\dagger}) &= \mathbb{I}.
\end{aligned}$$

This implies,

$$(B_0^0 \cdot B_1^{0\dagger} + B_1^0 \cdot B_0^{0\dagger}) = 0.$$

Now $B_0 = (B_0^0 - B_1^0)$. From this we can get,

$$\begin{aligned}
B_0 B_0^\dagger &= (B_0^0 - B_1^0)(B_0^0 - B_1^0)^\dagger \\
&= B_0^0 \cdot B_0^{0\dagger} - B_0^0 \cdot B_1^{0\dagger} - B_1^0 \cdot B_0^{0\dagger} + B_1^0 \cdot B_1^{0\dagger} \\
&= (B_0^0 + B_1^0) - (B_0^0 \cdot B_1^{0\dagger} + B_1^0 \cdot B_0^{0\dagger}) \\
&= \mathbb{I} + 0 = \mathbb{I}.
\end{aligned}$$

Similarly, it can be shown that, $B_1 B_1^\dagger = B_1^\dagger B_1 = \mathbb{I}$.

So, from equation 4.48, we can write that for unitary observables B_0 and B_1 ,

$$\begin{aligned}
|\{B_0, B_1\}|^2 + |[B_0, B_1]|^2 &= 2(B_0 B_1)^\dagger (B_0 B_1) \\
&\quad + 2(B_1 B_0)^\dagger (B_1 B_0) \\
&= 2\mathbb{I} + 2\mathbb{I} = 4\mathbb{I}.
\end{aligned}$$

This implies,

$$|\{B_0, B_1\}| = \sqrt{4\mathbb{I} - |[B_0, B_1]|^2}.$$

So, the simplified expression of T_α will be of the form

$$T_\alpha = (\alpha^2 - 1)\sqrt{4\mathbb{I} - |[B_0, B_1]|^2} + 2\alpha|[B_0, B_1]|.$$

This is the maximum value of T_α and here T_α attains this maximum value because of projective observables. Now one can easily check that the value of $|[B_0, B_1]|$ which maximizes the value of T_α is $|[B_0, B_1]| = \frac{4\alpha}{(\alpha^2+1)}\mathbb{I}$ and the corresponding value of T_α is $2(\alpha^2 + 1)\mathbb{I}$. This implies that,

$$T_\alpha = 2(\alpha^2 + 1)\mathbb{I}.$$

From this value of T_α and from the expression of $W_{\mathcal{A}}^2$ mentioned in equation 4.47, we can easily write that the value of $W_{\mathcal{A}}$ is upper bounded by the following quantity.

$$W_{\mathcal{A}} \leq \sqrt{2(\alpha^2 + 1)\mathbb{I} \otimes \mathbb{I} + T_\alpha \otimes \mathbb{I}}. \quad (4.49)$$

where $T_\alpha = 2(\alpha^2 + 1)\mathbb{I}$.

Now, the value β obtained in OBStest of our algorithm can be written alternatively as $\beta = \frac{\text{Tr}(W_{\mathcal{A}}\rho_{\mathcal{AB}})}{4}$ where $\rho_{\mathcal{AB}}$ is the density matrix representation of the shared states $|\phi\rangle_{\mathcal{AB}}$ i.e., $\rho_{\mathcal{AB}} = |\phi\rangle_{\mathcal{AB}}\langle\phi|$. From this expression of β , one can easily derive that the value of β^2 is upper bounded by the following quantity.

$$\beta^2 \leq \frac{\text{Tr}(W_{\mathcal{A}}^2\rho_{\mathcal{AB}})}{16}.$$

Now if we assume $t_\alpha := \frac{1}{4}\text{Tr}(T_\alpha\rho_{\mathcal{B}}) - \frac{1}{2}(\alpha^2 - 1)$ (where $\rho_{\mathcal{B}}$ is the reduced state at Bob's side) then using this value of t_α along with the value of $W_{\mathcal{A}}$ obtained from expression 4.49 and the upper bound on the value of β^2 , we can write that the β value mentioned in OBStest is upper bounded by the following quantity.

$$\beta \leq \frac{\sqrt{\alpha^2 + t_\alpha}}{2}. \quad (4.50)$$

Now here, the observables are projective (i.e., $B_j^2 = \mathbb{I}$) and the anti commutator $\{B_0, B_1\}$ is a positive semi definite operator. Since we have already shown that the value of the anti-hermitian operator $[[B_0, B_1]]$ is $[[B_0, B_1]] = \frac{4\alpha}{(\alpha^2+1)}\mathbb{I}$ for the maximum value of T_α , the spectral decomposition of $[B_0, B_1]$ can be written as,

$$[B_0, B_1] = \frac{4\alpha.i}{(\alpha^2 + 1)}(P_+ - P_-).$$

for some orthogonal projectors P_+ and P_- such that $(P_+ + P_-) = \mathbb{I}$. As it is well-known that for projective observables, the commutator holds the property $B_0[B_0, B_1]B_0 = -[B_0, B_1]$, we can easily conclude that $B_0P_\pm B_0 = P_\mp$. Let us consider that $\{|e_j^0\rangle\}_j$ is an orthonormal basis for the support of P_+ and $\{|e_j^1\rangle\}_j$ is an orthonormal basis for the support of P_- where $|e_j^1\rangle = B_0|e_j^0\rangle$. We define the unitary operator U_0 as

$$U_0|e_j^d\rangle = \frac{1}{\sqrt{2}}[|0\rangle + (-1)^d i|1\rangle]|j\rangle.$$

for $d \in \{0, 1\}$. Then we can easily verify that

$$U_0[B_0, B_1]U_0^\dagger = \frac{4\alpha.i}{(\alpha^2 + 1)}\sigma_Y \otimes \mathbb{I}.$$

Since $\{\mathbb{I}, \sigma_X, \sigma_Y, \sigma_Z\}$ constitute an operator basis for linear operators acting on \mathbb{C}^2 , without loss of generality we can write

$$U_0 B_0 U_0^\dagger = \mathbb{I} \otimes K_0 + \sigma_X \otimes K_x + \sigma_Y \otimes K_y + \sigma_Z \otimes K_z.$$

for some hermitian operator K_0, K_x, K_y, K_z . For projective observable B_0 , one can easily check that $\{B_0, [B_0, B_1]\} = 0$. This relation satisfies only when $K_0 = K_y = 0$. As $B_0^2 = \mathbb{I}$, K_x and K_z must satisfy the relation

$$K_x^2 + K_z^2 = \mathbb{I} \quad \text{and} \quad [K_x, K_z] = 0.$$

So, we can easily write K_x and K_z in the following form.

$$K_x = \sum_j \sin 2\gamma_j |j\rangle \langle j|$$

$$K_z = \sum_j \cos 2\gamma_j |j\rangle \langle j|.$$

for some angle γ_j and some orthonormal basis $\{|j\rangle\}$. This implies that

$$U_0 B_0 U_0^\dagger = \sigma_X \otimes K_x + \sigma_Z \otimes K_z$$

$$= \sum_j (\sin 2\gamma_j \sigma_X + \cos 2\gamma_j \sigma_Z) \otimes |j\rangle \langle j|.$$

We now consider the following controlled unitary to align the qubit observables.

$$U_1 = \sum_j \exp(i\gamma_j \cdot \sigma_Y) \otimes |j\rangle \langle j|.$$

Now for this defined unitary operator, one can easily check that

$$U_1 U_0 B_0 U_0^\dagger U_1^\dagger = \sigma_Z \otimes \mathbb{I}$$

$$U_1 U_0 [B_0, B_1] U_0^\dagger U_1^\dagger = \frac{4\alpha \cdot i}{(\alpha^2 + 1)} \sigma_Y \otimes \mathbb{I}.$$

Like observable B_0 , an analogous reasoning can also be applied for observable B_1 and from that, without loss of generality we can write

$$U_1 U_0 B_1 U_0^\dagger U_1^\dagger = \sigma_X \otimes K'_x + \sigma_Z \otimes K'_z.$$

Since the commutators are positive semi definite and the observables are projective, we can easily check that

$$\begin{aligned}\{B_0, B_1\} &= |\{B_0, B_1\}| = \sqrt{4.\mathbb{I} - |[B_0, B_1]|^2} \\ &= \frac{2(\alpha^2 - 1)}{(\alpha^2 + 1)}.\mathbb{I}.\end{aligned}$$

Now we define $2\theta := \arcsin\left(\frac{\alpha^2-1}{\alpha^2+1}\right) \in [0, \frac{\pi}{2}]$. From this relation, imposing consistency on the anti commutator, we get,

$$K'_z = \sin 2\theta.\mathbb{I}.$$

On the other hand, imposing consistency on the commutator, we get,

$$K'_x = \cos 2\theta.\mathbb{I}.$$

Now, from the relation $2\theta := \arcsin\left(\frac{\alpha^2-1}{\alpha^2+1}\right)$, we can get the value of α which is

$$\alpha = \frac{(\cos \theta + \sin \theta)}{|(\cos \theta - \sin \theta)|}.$$

For this value of α , we can easily derive that $t_\alpha = 1$. This implies that the simplified expression for β is,

$$\beta = \frac{\sqrt{1 + \alpha^2}}{2}. \quad (4.51)$$

where $\alpha = \frac{(\cos \theta + \sin \theta)}{|(\cos \theta - \sin \theta)|}$. Now from this value of α , we can derive the value of $\sqrt{1 + \alpha^2}$ which is,

$$\sqrt{1 + \alpha^2} = \frac{\sqrt{2}}{|(\cos \theta - \sin \theta)|}. \quad (4.52)$$

So, the value of β corresponding to these observables B_0 and B_1 will be,

$$\beta = \frac{1}{\sqrt{2}|(\cos \theta - \sin \theta)|}. \quad (4.53)$$

If we consider $U_{\mathcal{B}} = U_0^\dagger U_1^\dagger$ then the observables B_0 and B_1 will be of the form

$$\begin{aligned}B_0 &= U_{\mathcal{B}}(\sigma_Z \otimes \mathbb{I})U_{\mathcal{B}}^\dagger \\ B_1 &= U_{\mathcal{B}}(\cos 2\theta\sigma_X + \sin 2\theta\sigma_Z \otimes \mathbb{I})U_{\mathcal{B}}^\dagger.\end{aligned}$$

This implies that in the OBStest, if β is equal to $\frac{1}{\sqrt{2}|\cos\theta - \sin\theta|}$, then the corresponding observables of Bob are same as the one described in the OBStest. This concludes the proof.

4.6 Verification of Alice's POVM elements

In the QPQ protocol, Alice needs to make sure her measurement device works properly, i.e., she should be able to distinguish between $|0\rangle$ ($|1\rangle$) and $|0'\rangle$ ($|1'\rangle$) with certainty for (on average) around $(1 - \cos\theta)$ fraction of instances, where, $|0'\rangle = \cos\theta|0\rangle + \sin\theta|1\rangle$ ($|1'\rangle = \sin\theta|0\rangle - \cos\theta|1\rangle$). Let $M^0 = \{M_0^0, M_1^0, M_2^0\}$ ($M^1 = \{M_0^1, M_1^1, M_2^1\}$) the set of Alice's POVMs, which distinguishes the states $\{|0\rangle, |0'\rangle\}$ ($\{|1\rangle, |1'\rangle\}$). Here we show that if the input states are of the form $|0\rangle$ ($|1\rangle$) or $|0'\rangle$ ($|1'\rangle$) and Alice manages to distinguish the states with certainty for (on average) around $(1 - \cos\theta)$ fraction of instances then $M_i^0 = D_i^0$ ($M_i^1 = D_i^1$) for $i \in \{0, 1, 2\}$. In order to prove this, here we first represent the interactions between Bob and Alice in the proposed DI-QPQ protocol in the form of a game, called $\text{POVMgame}(M^y, y)$ for better understanding, where the agent A_1 represents Bob and the agent A_2 represents Alice. The game is as follows,

<p>Algorithm 5: $\text{POVMgame}(M^y, y)$</p> <ul style="list-style-type: none"> • A_1 declares y whenever the state at his side (and also at A_2's side) is either ρ_x^y or $\rho_{x\oplus 1}^y$ for the randomly chosen x values (i.e., for $x \in_R \{0, 1\}$), where $\rho_0^0 = 0\rangle\langle 0$, $\rho_1^0 = 0'\rangle\langle 0'$, $\rho_0^1 = 1\rangle\langle 1$ and $\rho_1^1 = 1'\rangle\langle 1'$. • A_2 measures her state (which is either ρ_x^y or $\rho_{x\oplus 1}^y$) using the POVM M^y (where $M^y = \{M_0^y, M_1^y, M_2^y\}$) and sends the outcome $b \in \{0, 1, 2\}$ to A_1. • A_2 wins if and only if, $\Omega^y = \sum_{b,x \in \{0,1\}} (-1)^{b\oplus x} \text{Tr}[M_b^y \rho_x^y] = \frac{2\sin^2\theta}{1+\cos\theta}$.
--

Theorem 6. *In $\text{POVMgame}(M^y, y)$, if A_1 chooses $y = 0$ and the states at A_2 's end are $\rho_0^0 = |0\rangle\langle 0|$ and $\rho_1^0 = |0'\rangle\langle 0'|$ and if A_2 manages to win the game, i.e., $\Omega^0 = \frac{2\sin^2\theta}{1+\cos\theta}$, then this implies, A_2 's measurement devices are of the following form (up to a local unitary).*

$$M_0^0 = \frac{1}{(1 + \cos\theta)} (|1'\rangle\langle 1'|) \quad (4.54)$$

$$M_1^0 = \frac{1}{(1 + \cos\theta)} (|1\rangle\langle 1|) \quad (4.55)$$

$$M_2^0 = \mathbb{I} - M_0^0 - M_1^0. \quad (4.56)$$

where, $|1'\rangle = \sin\theta|0\rangle - \cos\theta|1\rangle$.

Proof. In the POVMgame(M^y, y), A_2 applies M^0 on a single qubit state ρ_x^0 (where $x \in_R \{0, 1\}$). So, without any loss of generality we can assume that $M_i^0 \in M^0$ has the following form.

$$M_i^0 = \lambda_i^0(\mathbb{I} + \vec{m}_i^0 \cdot \vec{\sigma}). \quad (4.57)$$

where $\vec{m}_i^0 = [m_{i0}^0, m_{i1}^0, m_{i2}^0]$ and it is the Bloch vector with length at most one, $\vec{\sigma} = [\sigma_X, \sigma_Y, \sigma_Z]$ are the Pauli matrices and $\lambda_i \geq 0$. In this case, one may wonder how we can fix the dimension of M_i^0 here in the proof in DI scenario. The answer to this question is that here we are able to fix the dimension of M_i^0 and choose this particular general form because of the tests mentioned earlier in the source device verification phase (corresponding result mentioned in Corollary 2) and DI testing phase for Bob's measurement device (corresponding result mentioned in Theorem 2) which certifies that the states shared between Alice and Bob are EPR pairs (up to a unitary) and after Bob's projective measurements, the reduced states at Alice's side are one qubit states. Now, the condition $\sum_{i=0}^2 M_i^0 = \mathbb{I}$ leads us to the following relations.

$$\sum_{i=0}^2 \lambda_i^0 = 1 \quad (4.58)$$

$$\sum_{i=0}^2 \lambda_i^0 \vec{m}_i^0 = 0. \quad (4.59)$$

In terms of Bloch vector we can rewrite ρ_0^0, ρ_1^0 in following way.

$$\rho_0^0 = \frac{1}{2}(\mathbb{I} + \sigma_Z) \quad (4.60)$$

$$\rho_1^0 = \frac{1}{2}(\mathbb{I} + \sin 2\theta \sigma_X + \cos 2\theta \sigma_Z). \quad (4.61)$$

In the POVMgame(M^y, y) if A_2 would like to maximize her winning probability then she needs to maximize the following expression.

$$\Omega^0 = \sum_{b,x \in \{0,1\}} (-1)^{b \oplus x} \text{Tr}[M_b^0 \rho_x^0]. \quad (4.62)$$

In terms of $\lambda_i^0, \vec{m}_i^0, \vec{\sigma}$ we have,

$$\begin{aligned} \text{Tr}[M_0^0 \rho_0^0] &= \lambda_0^0(1 + m_{02}^0) \\ \text{Tr}[M_0^0 \rho_1^0] &= \lambda_0^0(1 + m_{00}^0 \sin 2\theta + m_{02}^0 \cos 2\theta) \\ \text{Tr}[M_1^0 \rho_0^0] &= \lambda_1^0(1 + m_{12}^0) \\ \text{Tr}[M_1^0 \rho_1^0] &= \lambda_1^0(1 + m_{10}^0 \sin 2\theta + m_{12}^0 \cos 2\theta). \end{aligned}$$

In terms of $\lambda_i^0, \vec{m}_i^0, \vec{\sigma}$ can rewrite Ω^0 as,

$$\begin{aligned}\Omega^0 &= \lambda_0^0(1 + m_{02}^0) + \lambda_1^0(1 + m_{10}^0 \sin 2\theta + m_{12}^0 \cos 2\theta) \\ &\quad - \lambda_0^0(1 + m_{00}^0 \sin 2\theta + m_{02}^0 \cos 2\theta) - \lambda_1^0(1 + m_{12}^0).\end{aligned}\tag{4.63}$$

As both $\text{Tr}[M_0^0 \rho_1^0]$ and $\text{Tr}[M_1^0 \rho_0^0]$ are positive quantity, hence

$$\Omega^0 \leq \lambda_0^0(1 + m_{02}^0) + \lambda_1^0(1 + m_{10}^0 \sin 2\theta + m_{12}^0 \cos 2\theta).\tag{4.64}$$

and this implies,

$$(1 + m_{00}^0 \sin 2\theta + m_{02}^0 \cos 2\theta) = 0\tag{4.65}$$

$$(1 + m_{12}^0) = 0.\tag{4.66}$$

According to the equation 4.66 we have $m_{12}^0 = -1$. As both of ρ_0^0, ρ_1^0 lie on the XZ plane and due to the freedom of local unitary without loss of generality we can assume $m_{01}^0 = m_{11}^0 = m_{21}^0 = 0$. Due to the positivity constraint ($M_i^0 \geq 0$) we have,

$$m_{00}^{0\ 2} + m_{02}^{0\ 2} \leq 1\tag{4.67}$$

$$m_{10}^{0\ 2} + m_{12}^{0\ 2} \leq 1\tag{4.68}$$

$$m_{20}^{0\ 2} + m_{22}^{0\ 2} \leq 1.\tag{4.69}$$

By combining the constraint equation 4.66 with the equation 4.68 we get, $m_{10}^0 = 0$. Hence,

$$\vec{m}_1^0 = [0, 0, -1].\tag{4.70}$$

and by substituting the values of m_{10}^0, m_{12}^0 in equation 4.64 we get the following expression of Ω^0 .

$$\Omega^0 \leq \lambda_0^0(1 + m_{02}^0) + \lambda_1^0(1 - \cos 2\theta).\tag{4.71}$$

Note that the expression of Ω^0 maximizes when $\lambda_0^0, m_{02}^0, \lambda_1^0$ maximizes and from the constraint equation 4.67 we get that $m_{00}^{0\ 2} + m_{02}^{0\ 2} \leq 1$. Hence, without any loss of generality we can assume that for the maximum value of Ω^0 , $m_{00}^{0\ 2} + m_{02}^{0\ 2} = 1$. So, we can parameterize m_{00}^0, m_{02}^0 as $\sin \alpha, \cos \alpha$ ($0 \leq \alpha \leq 2\pi$). By substituting $m_{00}^0 = \sin \alpha, m_{02}^0 = \cos \alpha$ in equation 4.65 we get,

$$1 + \sin \alpha \sin 2\theta + \cos \alpha \cos 2\theta = 0.$$

This implies,

$$\cos(\alpha - 2\theta) = -1.$$

As $0 \leq \alpha \leq 2\pi$, so $\cos(\alpha - 2\theta) = -1$ this implies,

$$\begin{aligned}\alpha - 2\theta &= \pi \quad \text{and,} \\ \alpha &= \pi + 2\theta.\end{aligned}\tag{4.72}$$

From the equation 4.72 we get,

$$\vec{m}_0^0 = [-\sin 2\theta, 0, -\cos 2\theta].\tag{4.73}$$

By substituting the expression of \vec{m}_0 in equation 4.71 we get,

$$\Omega^0 \leq (\lambda_0^0 + \lambda_1^0)(1 - \cos 2\theta).\tag{4.74}$$

By substituting the values of \vec{m}_0^0, \vec{m}_1^0 in equation 4.59 we get,

$$\lambda_2^0 m_{22}^0 - \lambda_0^0 \cos 2\theta = \lambda_1^0\tag{4.75}$$

$$\lambda_2^0 m_{20}^0 = \lambda_0^0 \sin 2\theta.\tag{4.76}$$

Due to the constraint equation 4.69, similar to \vec{m}_0^0 , here we parameterize the expression of m_{20}^0, m_{22}^0 as $\sin \beta, \cos \beta$ respectively. By substituting $m_{20}^0 = \sin \beta$ and $m_{22}^0 = \cos \beta$ in the equations 4.75 and 4.76 we get,

$$\lambda_2^0 \cos \beta - \lambda_0^0 \cos 2\theta = \lambda_1^0\tag{4.77}$$

$$\lambda_2^0 \sin \beta = \lambda_0^0 \sin 2\theta.\tag{4.78}$$

By solving equation 4.77 and equation 4.78 together with equation 4.58 we get,

$$\lambda_0^0 = \frac{\sin \beta}{\sin \beta + \sin 2\theta + \sin(2\theta - \beta)}\tag{4.79}$$

$$\lambda_1^0 = \frac{\sin(2\theta - \beta)}{\sin \beta + \sin 2\theta + \sin(2\theta - \beta)}.\tag{4.80}$$

Hence,

$$\lambda_0^0 + \lambda_1^0 = \frac{\sin \beta + \sin(2\theta - \beta)}{\sin \beta + \sin 2\theta + \sin(2\theta - \beta)}\tag{4.81}$$

$$= \frac{\cos(\theta - \beta)}{\cos \theta + \cos(\theta - \beta)}.\tag{4.82}$$

According to equation 4.74, for getting a tight upper bound on Ω^0 we need to

maximize $(\lambda_0^0 + \lambda_1^0)$. By equating $\frac{d(\lambda_0^0 + \lambda_1^0)}{d\beta} = 0$ in equation 4.82 we get,

$$\frac{\sin(\theta - \beta) \cos \theta}{\cos \theta + \cos(\theta - \beta)} = 0. \quad (4.83)$$

This implies,

$$\beta = \theta. \quad (4.84)$$

It is also easy to check that for $\theta = \beta$, the expression $\frac{d^2(\lambda_0^0 + \lambda_1^0)}{d\beta^2} < 0$. Hence, the expression $\lambda_0^0 + \lambda_1^0$ maximizes at the point $\beta = \theta$. Substituting this relation in equations 4.79 and 4.80 we get,

$$\lambda_0^0 = \lambda_1^0 = \frac{1}{2(1 + \cos \theta)}. \quad (4.85)$$

By substituting the values of $\lambda_0^0 + \lambda_1^0$ in equation 4.58 we get,

$$\lambda_2^0 = \frac{\cos \theta}{1 + \cos \theta}. \quad (4.86)$$

Hence, we get,

$$\Omega^0 \leq \frac{2 \sin^2 \theta}{1 + \cos \theta}. \quad (4.87)$$

and

$$M_0^0 = \frac{1}{2(1 + \cos \theta)} (\mathbb{I} - \sin 2\theta \sigma_X - \cos 2\theta \sigma_Z) \quad (4.88)$$

$$M_1^0 = \frac{1}{2(1 + \cos \theta)} (\mathbb{I} - \sigma_Z) \quad (4.89)$$

$$M_2^0 = \frac{\cos \theta}{1 + \cos \theta} (\mathbb{I} + \sin \theta \sigma_X + \cos \theta \sigma_Z). \quad (4.90)$$

We can rewrite the above expressions as follows,

$$M_0^0 = \frac{1}{(1 + \cos \theta)} (|1'\rangle\langle 1'|)$$

$$M_1^0 = \frac{1}{(1 + \cos \theta)} (|1\rangle\langle 1|)$$

$$M_2^0 = \mathbb{I} - M_0^0 - M_1^0.$$

where $|1'\rangle = \sin \theta |0\rangle - \cos \theta |1\rangle$. This concludes the proof. □

Similarly for the input states $|1\rangle, |1'\rangle$, one can conclude the following.

Theorem 7. In $\text{POVMgame}(M^y, y)$, if A_1 chooses $y = 1$ and the states at A_2 's end are $\rho_0^1 = |1\rangle\langle 1|$ and $\rho_1^1 = |1'\rangle\langle 1'|$ and if A_2 manages to win the game, i.e., $\Omega^1 = \frac{\sin^2 \theta}{1 + \cos \theta}$, then this implies, A_2 's measurement devices are of the following form (up to a local unitary).

$$M_0^1 = \frac{1}{(1 + \cos \theta)}(|0'\rangle\langle 0'|) \quad (4.91)$$

$$M_1^1 = \frac{1}{(1 + \cos \theta)}(|0\rangle\langle 0|) \quad (4.92)$$

$$M_2^1 = \mathbb{I} - M_0^1 - M_1^1. \quad (4.93)$$

where $|0'\rangle = \cos \theta|0\rangle + \sin \theta|1\rangle$.

Proof. In the $\text{POVMgame}(M^y, y)$, A_2 applies M^1 on a single qubit state ρ_x^1 (where $x \in_R \{0, 1\}$). So, without any loss of generality we can assume that $M_i^1 \in M^1$ has the following form.

$$M_i^1 = \lambda_i^1(\mathbb{I} + \vec{m}_i^1 \cdot \vec{\sigma}). \quad (4.94)$$

where $\vec{m}_i^1 = [m_{i0}^1, m_{i1}^1, m_{i2}^1]$ and it is the Bloch vector with length at most one, $\vec{\sigma} = [\sigma_X, \sigma_Y, \sigma_Z]$ are the Pauli matrices and $\lambda_i^1 \geq 0$. The condition $\sum_{i=0}^2 M_i^1 = \mathbb{I}$ leads us to the following relations.

$$\sum_{i=0}^2 \lambda_i^1 = 1 \quad (4.95)$$

$$\sum_{i=0}^2 \lambda_i^1 \vec{m}_i^1 = 0. \quad (4.96)$$

In terms of Bloch vector we can rewrite ρ_0^1, ρ_1^1 in following way.

$$\rho_0^1 = \frac{1}{2}(\mathbb{I} - \sigma_Z) \quad (4.97)$$

$$\rho_1^1 = \frac{1}{2}(\mathbb{I} - \sin 2\theta \sigma_X - \cos 2\theta \sigma_Z). \quad (4.98)$$

In the $\text{POVMgame}(M^y, y)$ if A_2 would like to maximize her winning probability then she needs to maximize the following expression.

$$\Omega^1 = \sum_{b,x \in \{0,1\}} (-1)^{b \oplus x} \text{Tr}[M_b^1 \rho_x]. \quad (4.99)$$

In terms of $\lambda_i^1, \vec{m}_i^1, \vec{\sigma}$ we have,

$$\begin{aligned}
\text{Tr}[M_0^1 \rho_0^1] &= \lambda_0^1(1 - m_{02}^1) \\
\text{Tr}[M_0^1 \rho_1^1] &= \lambda_0^1(1 - m_{00}^1 \sin 2\theta - m_{02}^1 \cos 2\theta) \\
\text{Tr}[M_1^1 \rho_0^1] &= \lambda_1^1(1 - m_{12}^1) \\
\text{Tr}[M_1^1 \rho_1^1] &= \lambda_1^1(1 - m_{10}^1 \sin 2\theta - m_{12}^1 \cos 2\theta).
\end{aligned}$$

In terms of $\lambda_i^1, \vec{m}_i^1, \vec{\sigma}$ can rewrite Ω^1 as,

$$\begin{aligned}
\Omega^1 &= \lambda_0^1(1 - m_{02}^1) + \lambda_1^1(1 - m_{10}^1 \sin 2\theta - m_{12}^1 \cos 2\theta) \\
&\quad - \lambda_0^1(1 - m_{00}^1 \sin 2\theta - m_{02}^1 \cos 2\theta) - \lambda_1^1(1 - m_{12}^1).
\end{aligned} \tag{4.100}$$

As both $\text{Tr}[M_0^1 \rho_1^1]$ and $\text{Tr}[M_1^1 \rho_0^1]$ are positive quantity, hence

$$\Omega^1 \leq \lambda_0^1(1 - m_{02}^1) + \lambda_1^1(1 - m_{10}^1 \sin 2\theta - m_{12}^1 \cos 2\theta). \tag{4.101}$$

and this implies,

$$(1 - m_{00}^1 \sin 2\theta - m_{02}^1 \cos 2\theta) = 0 \tag{4.102}$$

$$(1 - m_{12}^1) = 0. \tag{4.103}$$

According to the equation 4.103 we have $m_{12}^1 = 1$. As both of ρ_0^1, ρ_1^1 lie on the XZ plane and due to the freedom of local unitary without loss of generality we can assume $m_{01}^1 = m_{11}^1 = m_{21}^1 = 0$. Due to the positivity constraint ($M_i^1 \geq 0$) we have,

$$m_{00}^{1\ 2} + m_{02}^{1\ 2} \leq 1 \tag{4.104}$$

$$m_{10}^{1\ 2} + m_{12}^{1\ 2} \leq 1 \tag{4.105}$$

$$m_{20}^{1\ 2} + m_{22}^{1\ 2} \leq 1. \tag{4.106}$$

By combining the constraint equation 4.103 with the equation 4.105 we get, $m_{10}^1 = 0$. Hence,

$$\vec{m}_1^1 = [0, 0, 1]. \tag{4.107}$$

and by substituting the values of m_{10}^1, m_{12}^1 in equation 4.101 we get the following expression of Ω^1 .

$$\Omega^1 \leq \lambda_0^1(1 - m_{02}^1) + \lambda_1^1(1 - \cos 2\theta). \tag{4.108}$$

Note that the expression of Ω^1 maximizes when λ_0^1, λ_1^1 maximizes and m_{02}^1 minimizes and from the constraint equation 4.67 we get that $m_{00}^{1\ 2} + m_{02}^{1\ 2} \leq 1$. Hence, without any loss of generality we can assume that for the maximum value of Ω^1 , $m_{00}^{1\ 2} + m_{02}^{1\ 2} = 1$. So, we can parameterize m_{00}^1, m_{02}^1 as $\sin \alpha, \cos \alpha$ ($0 \leq \alpha \leq 2\pi$). By

substituting $m_{00}^1 = \sin \alpha$, $m_{02}^1 = \cos \alpha$ in equation 4.65 we get,

$$1 - \sin \alpha \sin 2\theta - \cos \alpha \cos 2\theta = 0.$$

This implies,

$$\cos(\alpha - 2\theta) = 1.$$

As $0 \leq \alpha \leq 2\pi$, so $\cos(\alpha - 2\theta) = 1$ this implies,

$$\begin{aligned} \alpha - 2\theta &= 0 \quad \text{or} \quad 2\pi \quad \text{and,} \\ \alpha &= 2\theta \quad \text{or} \quad (2\pi + 2\theta). \end{aligned} \tag{4.109}$$

One can easily check that for both these values of α , the value of m_{00}^1 and m_{02}^1 are $\sin 2\theta$ and $\cos 2\theta$ respectively. From the equation 4.109 we get,

$$\vec{m}_0^1 = [\sin 2\theta, 0, \cos 2\theta]. \tag{4.110}$$

By substituting the expression of \vec{m}_0^1 in equation 4.108 we get,

$$\Omega^1 \leq (\lambda_0^1 + \lambda_1^1)(1 - \cos 2\theta). \tag{4.111}$$

By substituting the values of \vec{m}_0^1, \vec{m}_1^1 in equation 4.96 we get,

$$\lambda_2^1 m_{22}^1 + \lambda_0^1 \cos 2\theta + \lambda_1^1 = 0 \tag{4.112}$$

$$\lambda_2^1 m_{20}^1 + \lambda_0^1 \sin 2\theta = 0. \tag{4.113}$$

Due to the constraint equation 4.106, similar to \vec{m}_0^1 , here we parameterize the expression of m_{20}^1, m_{22}^1 as $\sin \beta, \cos \beta$ respectively. By substituting $m_{20}^1 = \sin \beta$ and $m_{22}^1 = \cos \beta$ in the equations 4.112 and 4.113 we get,

$$\lambda_2^1 \cos \beta + \lambda_0^1 \cos 2\theta + \lambda_1^1 = 0 \tag{4.114}$$

$$\lambda_2^1 \sin \beta + \lambda_0^1 \sin 2\theta = 0. \tag{4.115}$$

By solving equation 4.114 and equation 4.115 together with equation 4.95 we get,

$$\lambda_0^1 = \frac{\sin \beta}{\sin \beta + \sin(2\theta - \beta) - \sin 2\theta} \tag{4.116}$$

$$\lambda_1^1 = \frac{\sin(2\theta - \beta)}{\sin \beta + \sin(2\theta - \beta) - \sin 2\theta}. \tag{4.117}$$

Hence,

$$\lambda_0^1 + \lambda_1^1 = \frac{\sin \beta + \sin(2\theta - \beta)}{\sin \beta + \sin(2\theta - \beta) - \sin 2\theta} \quad (4.118)$$

$$= \frac{\cos(\theta - \beta)}{\cos(\theta - \beta) - \cos \theta}. \quad (4.119)$$

According to equation 4.111, for getting a tight upper bound on Ω^1 we need to maximize $(\lambda_0^1 + \lambda_1^1)$. By equating $\frac{d(\lambda_0^1 + \lambda_1^1)}{d\beta} = 0$ in equation 4.119 we get,

$$\frac{-\sin(\theta - \beta) \cos \theta}{\cos \theta + \cos(\theta - \beta)} = 0. \quad (4.120)$$

This implies,

$$\text{either } \beta = \theta \text{ or } (\theta - \beta) = \pi. \quad (4.121)$$

Now, one can easily check that for $\theta = \beta$, the eigen value of M_2^1 becomes negative which is not possible. So, the solution here is $(\theta - \beta) = \pi$. One can also check that for $(\theta - \beta) = \pi$, the expression $\frac{d^2(\lambda_0^1 + \lambda_1^1)}{d\beta^2} < 0$. Hence, the expression $\lambda_0^1 + \lambda_1^1$ maximizes at the point $(\theta - \beta) = \pi$. Substituting this relation in equations 4.116 and 4.117 we get,

$$\lambda_0^1 = \lambda_1^1 = \frac{1}{2(1 + \cos \theta)}. \quad (4.122)$$

By substituting the values of $\lambda_0^1 + \lambda_1^1$ in equation 4.95 we get,

$$\lambda_2^1 = \frac{\cos \theta}{1 + \cos \theta}. \quad (4.123)$$

Hence, we get,

$$\Omega^1 \leq \frac{2 \sin^2 \theta}{1 + \cos \theta}. \quad (4.124)$$

The corresponding measurement operators using which A_2 can achieve $\Omega^1 = \frac{2 \sin^2 \theta}{1 + \cos \theta}$ is given by,

$$M_0^1 = \frac{1}{2(1 + \cos \theta)} (\mathbb{I} + \sin 2\theta \sigma_X + \cos 2\theta \sigma_Z) \quad (4.125)$$

$$M_1^1 = \frac{1}{2(1 + \cos \theta)} (\mathbb{I} + \sigma_Z) \quad (4.126)$$

$$M_2^1 = \frac{\cos \theta}{1 + \cos \theta} (\mathbb{I} - \sin \theta \sigma_X - \cos \theta \sigma_Z). \quad (4.127)$$

We can rewrite the above expressions as follows,

$$\begin{aligned}
M_0^1 &= \frac{1}{(1 + \cos \theta)} (|0'\rangle\langle 0'|) \\
M_1^1 &= \frac{1}{(1 + \cos \theta)} (|0\rangle\langle 0|) \\
M_2^1 &= \mathbb{I} - M_0^1 - M_1^1.
\end{aligned}$$

where, $|0'\rangle = \cos \theta |0\rangle + \sin \theta |1\rangle$. This concludes the proof. □

From the results of theorem 6 and 7, it is clear that the success probability $(1 - \cos \theta)$ in distinguishing two non-orthogonal states $\{|0\rangle, |0'\rangle\}$ (or $\{|1\rangle, |1'\rangle\}$) can be achieved only when the chosen POVM's are of the specified form as chosen by Alice for the QPQ scheme. From the results mentioned in [62], one can easily conclude that $(1 - \cos \theta)$ is the optimal success probability that can be achieved in distinguishing two non-orthogonal states. So from these two results, one can easily conclude that Alice can get optimal number of raw key bits in this QPQ scheme.

4.7 Correctness of the scheme considering devices “up to a unitary”

In the device independent testing phases of our proposed scheme (i.e., in source device verification phase, Bob’s measurement device verification phase and Alice’s POVM device verification phase), the tests certify that the devices perform exactly same as that is mentioned in the proposed scheme or “up to a unitary” of the actual device. This implies that the source device supplies states that are exactly of the same form or “up to a unitary” (i.e., the states received after applying a unitary operation) of the original state and the measurement devices measure in exactly the same specified basis or “up to a unitary” (i.e., the measurement bases received after applying a unitary operation) of the actual basis.

Thus, because of this “up to unitary” deviation, it is necessary to check whether the protocol preserves its correctness condition whenever the devices are “up to unitary” of the actual devices.

Let us consider that the measurement devices of Alice and Bob perform measurements in the bases which are up to unitary U_2 such that

$$U_2 = \begin{bmatrix} a & b \\ -e^{i\phi}b^* & e^{i\phi}a^* \end{bmatrix}.$$

Where, $a, b \in \mathbb{C}$ such that $|a|^2 + |b|^2 = 1$ and ϕ is the relative angle. Let us also assume that the source device supplies states which are up to unitary U_4 where

$$U_4 = U_2 \otimes U_2.$$

This implies that the states supplied by the source device are of the form

$$U_4(\phi_{AB}) = \frac{1}{\sqrt{2}}[|00\rangle + e^{i\phi}(a^*b - ab^*)|01\rangle + e^{i\phi}(a^*b - ab^*)|10\rangle + e^{2i\phi}(a^{*2} + b^{*2})|11\rangle].$$

Bob's device measures in the basis $\{U_2|0\rangle, U_2|1\rangle\} = \{(a|0\rangle - e^{i\phi}b^*|1\rangle), (b|0\rangle + e^{i\phi}a^*|1\rangle)\}$ and $\{U_2|0'\rangle, U_2|1'\rangle\} = \{(a \cos \theta + b \sin \theta)|0\rangle + e^{i\phi}(a^* \sin \theta - b^* \cos \theta)|1\rangle, (a \sin \theta - b \cos \theta)|0\rangle - e^{i\phi}(a^* \cos \theta + b^* \sin \theta)|1\rangle\}$ instead of the basis $\{|0\rangle, |1\rangle\}$ and $\{|0'\rangle, |1'\rangle\}$ respectively. Alice's POVM devices are either $D^0 = \{D_0^0, D_1^0, D_2^0\}$ or $D^1 = \{D_0^1, D_1^1, D_2^1\}$ for $a_i = 0$ and $a_i = 1$ respectively where

$$\begin{aligned} D_0^0 &= \frac{1}{(1 + \cos \theta)}(U_2|1'\rangle \langle 1'| U_2^\dagger) \\ D_1^0 &= \frac{1}{(1 + \cos \theta)}(U_2|1\rangle \langle 1| U_2^\dagger) \\ D_2^0 &= \mathbb{I} - D_0^0 - D_1^0. \end{aligned}$$

and

$$\begin{aligned} D_0^1 &= \frac{1}{(1 + \cos \theta)}(U_2|0'\rangle \langle 0'| U_2^\dagger) \\ D_1^1 &= \frac{1}{(1 + \cos \theta)}(U_2|0\rangle \langle 0| U_2^\dagger) \\ D_2^1 &= \mathbb{I} - D_0^1 - D_1^1. \end{aligned}$$

One can easily check that whenever Bob measures in $\{U_2|0\rangle, U_2|1\rangle\}$ or $\{U_2|0'\rangle, U_2|1'\rangle\}$ basis randomly on his qubit of the shared state $U_4(\phi_{AB})$, the qubit at Alice's side will also collapse to $U_2|0\rangle$ or $U_2|1\rangle$ for the first case and $U_2|0'\rangle$ or $U_2|1'\rangle$ for the second case.

Now, if Alice chooses POVM device $D^0 = \{D_0^0, D_1^0, D_2^0\}$ for $a_i = 0$, the probabilities of getting different outcomes for two different input states are as follows-

$$\begin{aligned} \Pr(D_0^0|U_2|0\rangle) &= (1 - \cos \theta) \\ \Pr(D_1^0|U_2|0\rangle) &= 0 \\ \Pr(D_2^0|U_2|0\rangle) &= \cos \theta \\ \Pr(D_0^0|U_2|0'\rangle) &= 0 \\ \Pr(D_1^0|U_2|0'\rangle) &= (1 - \cos \theta) \\ \Pr(D_2^0|U_2|0'\rangle) &= \cos \theta \end{aligned}$$

Similarly, if Alice chooses POVM device $D^1 = \{D_0^1, D_1^1, D_2^1\}$ for $a_i = 1$, the

probabilities of getting different outcomes for two different input states are as follows-

$$\begin{aligned}
\Pr(D_0^1|U_2|1) &= (1 - \cos \theta) \\
\Pr(D_1^1|U_2|1) &= 0 \\
\Pr(D_2^1|U_2|1) &= \cos \theta \\
\Pr(D_0^1|U_2|1') &= 0 \\
\Pr(D_1^1|U_2|1') &= (1 - \cos \theta) \\
\Pr(D_2^1|U_2|1') &= \cos \theta
\end{aligned}$$

According to the protocol, whenever $a_i = 0$ and Alice gets $D_0^0(D_1^0)$, she outputs $r_{\mathcal{A}_i} = 0(1)$. Whenever, $a_i = 1$ and she gets $D_0^1(D_1^1)$, she outputs $r_{\mathcal{A}_i} = 0(1)$. So, in this case, the success probability of Alice to guess the i -th raw key bit r_i of Bob will be,

$$\begin{aligned}
&\Pr(r_{\mathcal{A}_i} = r_i) \\
&= \Pr(r_{\mathcal{A}_i} = 0, r_i = 0) + \Pr(r_{\mathcal{A}_i} = 1, r_i = 1) \\
&= (1 - \cos \theta).
\end{aligned}$$

This shows that whenever the devices (both source and measurement devices) involved in this scheme are “up to a unitary” of the original specified device, then also the proposed scheme satisfies the correctness condition.

4.8 Discussion and Conclusion

The initial QPQ schemes assumed trust in the devices involved, leading to security issues depending on device functionality. Maitra et al. [77] first introduced DI in the QPQ domain by proposing a semi-DI version of the QPQ scheme [117]. In this chapter, we move one step further and propose a novel fully DI-QPQ scheme using maximally entangled states (EPR Pairs) for improved robustness. Our scheme achieves the optimal number of raw key bits for client Alice. We analyze security in a general way against all attacks preserving correctness. We provide upper bounds on the cheating probabilities for both the dishonest client and server. This new QPQ scheme with the incorporation of QKD has the potential to become a crucial near-term application of the quantum internet.

Proposal For Fully Device Independent QPQ using Non-maximally Entangled States

As mentioned earlier, Maitra et al. [77] initially highlighted that the security of the existing QPQ schemes (up until that point) relied on trust assumptions regarding the devices involved, including the source and measurement devices. They specifically examined the proposal by Yang et al. [117] and demonstrated that if the source device, responsible for providing the shared states, does not function correctly, the client can retrieve more data bits than intended. To overcome this security loophole and remove the trustful assumptions over the devices, they suggested a Device Independent (DI) version of the QPQ scheme [117] in [77]. They introduced a local testing phase at the server side in [77] which certifies the measurement devices at the server side and the state generation device. However, this test does not certify the measurement devices at the client's side. So, their proposal in [77] is basically a semi-DI version of the QPQ scheme [117]. Although this limitation is mentioned in the previous chapter (i.e., in paper [15]), to the best of our knowledge, the procedure for proper DI certification of the QPQ scheme [117] is not mentioned anywhere.

In this direction, here in this chapter, we discuss about our work that overcomes the limitations in [77] and propose a full DI version of the Yang et al. scheme [117]. Our proposal exploits the proper self-testing mechanism of the observables involved in [117] along with a local version of the tiltedCHSH test to certify all the measurement devices. We also compare the performance of this proposed full DI version of the QPQ scheme [117] with the performance of the full DI-QPQ scheme mentioned in the previous chapter (i.e., in [15]) and discuss the relative advantages of both these protocols. We further come up with a DI proposal for a modification of [117] where the client can retrieve optimal raw key bits at her end. In opposition to current DI-QPQ approaches, here in this modified proposal, we replace the usual projective measurement at the client's side with the optimal POVM measurement to retrieve the maximum number of shared raw key bits. A flow diagram involving the evolution of the QPQ scheme [117] in the DI scenario is shown in Figure 5-1.

Before explaining our exact contributions in detail in Section 5.2, we first revisit the QPQ scheme [117] and its DI version in [77] in Section 5.1. Then we discuss

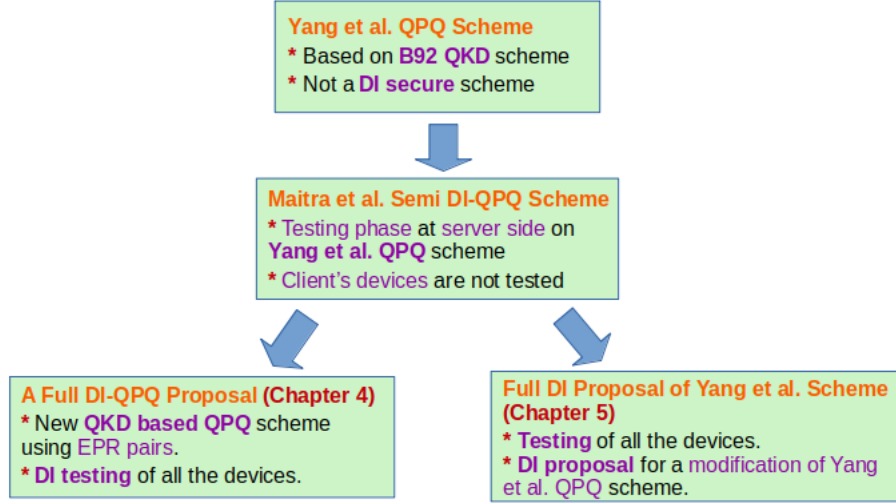


Figure 5-1: Evolution of QPQ schemes in DI scenario

an attack on the DI-QPQ scheme [77] in Section 5.3 and propose the full DI version of the QPQ scheme [117] in Section 5.4. We further propose a full DI version for a modification of [117] in Section 5.5 where the client can retrieve an optimal number of raw key bits during the oblivious key generation phase. Next, in the subsequent sections (i.e., in Sections 5.6, 5.7 and 5.8), we mention the detailed proofs of our results.

5.1 Revisiting the QPQ scheme [117] and its DI version in [77]

In this section, we first revisit the QPQ protocol mentioned in [117] and then restate the DI version of this QPQ scheme introduced in [77]. In [117], the authors proposed a QKD-based QPQ scheme exploiting the idea of B92 QKD protocol. Their proposed QPQ scheme is composed of mainly two phases namely the key generation phase and the private query phase.

In [117], non-maximally entangled states are shared between Bob and Alice which are of the form $\frac{1}{\sqrt{2}}(|0\rangle|\phi_0\rangle + |1\rangle|\phi_1\rangle)$ where $|\phi_0\rangle = (\cos \frac{\theta}{2}|0\rangle + \sin \frac{\theta}{2}|1\rangle)$ and $|\phi_1\rangle = (\cos \frac{\theta}{2}|0\rangle - \sin \frac{\theta}{2}|1\rangle)$ (at the beginning of the protocol, the exact value of this θ is decided by the server Bob to the third party based on the number of raw key bits that Bob wants Alice to know after the key generation phase). Bob first receives the states from the third party and then sends the second particle of each of those states to Alice. After receiving the particles, Alice announces all those instances where she receives the particles correctly, and then they discard all the rest instances where Alice does not receive the particles correctly. After post-selection, Bob measures each of his particles of the shared states in $\{|0\rangle, |1\rangle\}$ basis, and Alice measures each

of her particles either in $\{|\phi_0\rangle, |\phi_0^\perp\rangle\}$ basis or in $\{|\phi_1\rangle, |\phi_1^\perp\rangle\}$ basis randomly. Bob considers the raw key bit of a particular instance as 0 if he receives the outcome $|0\rangle$ and 1 otherwise. Similarly, if Alice receives $|\phi_0^\perp\rangle$ for a particular instance, then she concludes that the corresponding raw key bit at Bob's side is 1 and if she receives $|\phi_1^\perp\rangle$, she concludes that the corresponding raw key bit at Bob's side is 0. This implies that Alice can retrieve the raw key bits correctly only when she receives the outcome $|\phi_0^\perp\rangle$ or $|\phi_1^\perp\rangle$. After measurement, Alice and Bob perform classical post-processing over their raw key bits so that Alice's information about the final key reduces to one bit. This implies that after this key generation phase, Bob knows the entire key whereas Alice knows only some bits (more specifically one bit) of the final key.

In the private query phase, if Alice knows the j -th bit of the final key and wants to retrieve the bit indexed by i of the database then she declares the integer $s = (j - i)$ publicly. Bob then shifts his key by s bits, encrypts the database with this shifted key using the one-time pad, and sends it to Alice. Alice decrypts the j -th bit and gets the required element of the database.

It is already mentioned in [117] that by following the specified strategy, Alice can conclusively retrieve only $\frac{\sin^2 \theta}{2}$ (on average) fraction of bits of the entire raw key obtained by Bob. This guarantees the security of the proposed QPQ scheme because although Alice gets the whole encrypted database, she can not retrieve all the database bits because of her partial knowledge about the raw key as well as the final key.

However, it was shown in [77] that if the dishonest Alice colludes with the third party and supplies the states of the form $(\alpha|0\rangle|\phi_0\rangle + \beta|1\rangle|\phi_1\rangle)$ where $|\alpha|^2 = (\frac{1}{2} + \epsilon)$ and $|\beta|^2 = (\frac{1}{2} - \epsilon)$ then the dishonest Alice can retrieve additional $2\epsilon^2 \sin^2 \theta$ fraction of bits of the entire raw key. For this reason, to overcome the security loophole (a schematic diagram of different phases of the DI-QPQ scheme [77] is shown in Subfigure (a) of Figure 5-3), a DI version of the QPQ scheme [117] was proposed in [77].

In the DI scheme proposed in [77], the server Bob performs a tilted version of the original CHSH test locally to certify the devices. Although this local test certifies the states and Bob's measurement devices (for the specific measurement bases chosen in the test), this local test actually fails to provide any certification about Alice's measurement devices as those devices aren't involved in this test. This implies that the scheme mentioned in [77] is a semi-DI version of the Yang et al. [117] QPQ scheme. Here we overcome the limitation of this scheme [77] and propose a full DI version of the Yang et al. [117] QPQ scheme.

5.2 Contribution of this chapter

In this chapter, we focus on the Yang et al. [117] QPQ scheme that provides privacy for both the user and the database owner in a classical database search. While Maitra et al. [77] came up with a semi-DI proposal of the scheme for improved security, we present an improvement with a full DI version. The contributions of this chapter can be summarized as follows.

1. In the DI proposal [77], the server Bob locally performs a tilted CHSH test which

only involves the entangled states and his own measurement devices. Here, we show that the local test mentioned in [77] fails to preserve the data privacy of the database (as the client’s measurement devices are not tested in [77]), and the client Alice can retrieve some additional raw key bits (as well as the data bits in a single database query) if she performs an optimal POVM measurement at her side instead of the projective measurement mentioned in [117].

2. We propose a full DI version of the QPQ scheme [117] by exploiting a local version of the tiltedCHSH test (mentioned in [8, 14]) at both the server and the client’s side (in the *source device and Bob’s measurement device verification phase* of our scheme) along with the self-testing of projective measurements (mentioned in [65]) at the client Alice’s side (in the *Alice’s POVM device verification phase* of our proposal). The local test mentioned in [77] does not certify the functionality of the client Alice’s measurement device, and Alice also can not certify the shared states. Here we overcome these limitations and check the functionality of all the devices involved in the QPQ scheme [117]. We also compare this proposed full DI version with the full DI-QPQ scheme mentioned in [15] considering different parameters and show that both these schemes have some relative advantages.
3. We further came up with a full DI proposal for a modification of [117] where the client Alice can retrieve optimal raw key bits at her end during the oblivious key generation phase. In this improved proposal, we exploit the proper self-testing mechanism of a particular class of POVM device along with the local version of the tiltedCHSH test (mentioned in [8, 14]) and the self-testing of projective measurement operators (mentioned in [65]) to certify all the devices.

5.3 An attack on the DI-QPQ scheme [77]

In the DI-QPQ scheme [77], the server Bob first selects some entangled states (from the set of states that will be used for the QPQ scheme [117]) and performs a tilted version of the actual CHSH test locally to certify the states and the measurement devices involved in the QPQ scheme [117]. However, this local test does not certify Alice’s measurement devices as it only involves the entangled states and Bob’s measurement devices. This implies that if Alice performs some other measurement at her side instead of the actual projective measurement (mentioned in [117]), then the local test (mentioned in [77]) can not detect that.

Now suppose, for the QPQ scheme [117], Alice measures her qubits using the POVM $D = \{D_0, D_1, D_2\}$ instead of performing the projective measurements in $\{|\phi_0\rangle, |\phi_0^\perp\rangle\}$ or $\{|\phi_1\rangle, |\phi_1^\perp\rangle\}$ basis randomly where

$$\begin{aligned}
D_0 &\equiv \frac{(\sin \frac{\theta}{2}|0\rangle + \cos \frac{\theta}{2}|1\rangle)(\sin \frac{\theta}{2}\langle 0| + \cos \frac{\theta}{2}\langle 1|)}{(1 + \cos \theta)} \\
D_1 &\equiv \frac{(\sin \frac{\theta}{2}|0\rangle - \cos \frac{\theta}{2}|1\rangle)(\sin \frac{\theta}{2}\langle 0| - \cos \frac{\theta}{2}\langle 1|)}{(1 + \cos \theta)} \\
D_2 &\equiv I - D_0 - D_1
\end{aligned}$$

In this case, Alice can successfully pass the local CHSH test (at Bob's side mentioned in [77]) if Bob's measurement devices measure correctly in all the bases mentioned in algorithm 1 of [77] and the states are of the actual form. So, Alice and Bob proceed further for the QPQ scheme where Alice measures her qubits using the POVM $D = \{D_0, D_1, D_2\}$.

Now, in this case, whenever Alice gets the outcome D_0 , she concludes that Bob's measurement outcome for that instance is $|0\rangle$ and the raw key bit at Bob's side is 0. Similarly, whenever Alice gets the outcome D_1 , she concludes that Bob's measurement outcome for that instance is $|1\rangle$ and the raw key bit at Bob's end is 1. However, if Alice gets the outcome D_2 , she remains inconclusive about the value of the raw key bit at Bob's side.

We now calculate the success probability of Alice in guessing the raw key bits correctly. Let us assume that $\Pr(D_j|\phi_i\rangle)$ denotes the probability of getting the result D_j whenever the state at Alice's side is $|\phi_i\rangle$ i.e.,

$$\Pr(D_j|\phi_i\rangle) = \langle \phi_i | D_j | \phi_i \rangle.$$

This implies that whenever the state at Alice's side is $|\phi_0\rangle$, the success probabilities are

$$\begin{aligned}
\Pr(D_0|\phi_0\rangle) &= \langle \phi_0 | D_0 | \phi_0 \rangle \\
&= (1 - \cos \theta) \\
\Pr(D_1|\phi_0\rangle) &= \langle \phi_0 | D_1 | \phi_0 \rangle \\
&= 0 \\
\Pr(D_2|\phi_0\rangle) &= \langle \phi_0 | D_2 | \phi_0 \rangle \\
&= \cos \theta
\end{aligned}$$

Similarly, one can calculate the success probabilities whenever the state at Alice's side is $|\phi_1\rangle$. The following table (i.e., Table 5.1) shows all the conditional probabilities.

Cond. Probability of Alice			
Alice \ Bob	A= D_0	A= D_1	A= D_2
B= $ \phi_0\rangle$	$1 - \cos \theta$	0	$\cos \theta$
B= $ \phi_1\rangle$	0	$1 - \cos \theta$	$\cos \theta$

Table 5.1: Probabilities of Different POVM Outcomes

According to this strategy, whenever Alice gets the outcome $D_0(D_1)$, she concludes that the raw key bit at Bob's side is 0(1). Thus, the success probability of Alice in guessing Bob's i -th raw key bit can be written as

$$\begin{aligned} & \Pr(R_i = R_{\mathcal{A}_i}) \\ &= \Pr(R_i = 0, R_{\mathcal{A}_i} = 0) + \Pr(R_i = 1, R_{\mathcal{A}_i} = 1) \\ &= (1 - \cos \theta). \end{aligned}$$

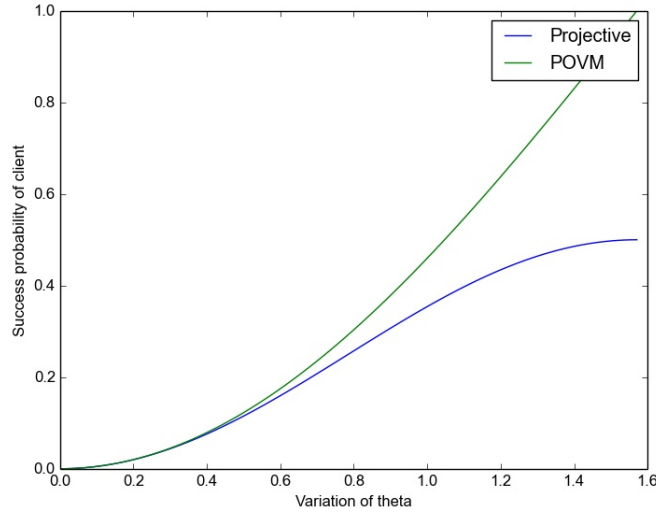


Figure 5-2: Comparison between the success probabilities of getting a raw key bit using projective and POVM measurements

The comparative study between this success probability (that can be achieved by Alice using the mentioned POVM) and the success probability of Alice in guessing a raw key bit correctly in the protocol [117] is shown in Figure 5-2. From the figure, one can easily check that the success probability using the mentioned POVM measurement outperforms the success probability using the projective measurements mentioned in [117] for all the values of θ .

It implies that whenever Bob's measurement device measures correctly in all the bases specified in [77] (algorithm 1), and the states in [117] are of the actual form, then Alice can successfully pass the testing phase in [77] even if she uses a different measurement device at her side and later can retrieve more number of raw key bits which violates the database privacy of the protocol [117].

Thus, our proposed attack is on the DI proposal in [77] which reveals the vulnerability of the scheme [77] and shows that the DI proposal in [77] fails to preserve the data privacy of the QPQ scheme [117].

To propose a full DI version of the scheme [117], a device certification test must be performed on all devices involved in the scheme.

5.4 Full DI proposal for the QPQ scheme [117]

In this section, we describe our proposal for certifying all the devices involved in the QPQ scheme [117]. We split up this entire section into two subsections. In the first subsection, we introduce different steps of our proposed scheme, and in the last subsection, we mention the security related issues of our proposal. Note that this proposal follows all the assumptions mentioned in Chapter 3 Section 3.6. Along with that, this proposal also assumes the following.

- For the QKD-based QPQ schemes, it is already shown in [63] that if the server attempts to retrieve more information about a client's query indices, then there is a risk of providing false information about the intended data bits to the client, which would damage the server's reputation as a database owner. That's why for the QPQ schemes, it is assumed that the server will not cheat if there exists a non-zero probability of being caught cheating. For this proposal, the server Bob can cheat without being detected because of the underlying computational hiding commitment scheme. But here, we assume that Bob has limitations on his computational resources and he is a polynomial time adversary i.e., Bob will try at most polynomial times to guess a committed value of the client Alice.

Note : For the QKD-based QPQ schemes, the size of the final key is equal to the size of the database which is usually very large, and the number of raw key bits is even more than that (usually some integer multiple of the number of final key bits). In this situation, it is impractical that the server spends more than the polynomial time to retrieve a raw key bit. For this reason, the polynomial time assumption seems justified here.

5.4.1 Proposed full DI version of the scheme [117]:

Depending on the functionality, our entire protocol is divided into four phases. The first phase is termed as *Source Device and Bob's Measurement Device Verification Phase*. This phase certifies that the states are of the specified form and Bob's device measures correctly on the specified basis. In this phase, Bob first receives all the states (that will be used for the protocol) from a third party (need not be a trusted one and may collude with the dishonest party) and shares those states with Alice. After that, they check the functionality of the devices in two subphases where at first Bob acts as a referee, chooses some samples randomly, and performs a tilted version of the original CHSH test locally to certify the states and his devices. In the next subphase, Alice also does the same that Bob did in the previous subphase and certifies the states.

After the certification of this source device and Bob's measurement device, they proceed to *Alice's Measurement Device Verification Phase*. This phase certifies the measurement bases of Alice specified in [117]. In this phase, Alice and Bob consider the remaining shared states and perform some measurements assuming their devices as unknown boxes. Then from the outcomes, Alice concludes about the functionality of her measurement device for those specified bases. After successful completion of

these two testing phases, Bob and Alice conclude that the states given to them are of the specified form and their measurement devices measure correctly in the bases specified in [117].

The next phase is termed *Key Generation Phase* where Bob generates a key and Alice knows some bits of that key and Bob can not guess the known indices of Alice. The last phase is termed as *private query phase* where Bob encrypts the entire database using the key generated in the previous phase and sends it to Alice. Alice then decrypts the intended bits of the database using her partial knowledge about the final key bits.

Our scheme consists of several steps, which are described below. It should be noted that channel noise is not considered in this description, so it is assumed that all operations are error-free.

Source Device and Bob's Measurement Device Verification Phase:

1. Bob starts with \mathcal{K} (we assume here that \mathcal{K} is asymptotically large) number of states (say $|\psi\rangle_{\mathcal{B}\mathcal{A}}$) provided by the third party and shares those states with Alice in such a way that the first particle of each state corresponds to Bob and the second particle corresponds to Alice.
2. Bob chooses $\frac{\gamma_1 \mathcal{K}}{2}$ instances randomly from these \mathcal{K} shared states (in practice, how Bob and Alice choose the specific value of γ_1 from the set $[0, 1]$ is mentioned in Section 4.4 of Chapter 4), declares those instances publicly and constructs the set $\Gamma_{\text{CHSH}}^{\mathcal{B}}$ with these chosen instances.
3. For all the instances in $\Gamma_{\text{CHSH}}^{\mathcal{B}}$, Alice sends her qubits to Bob.
4. For the instances in $\Gamma_{\text{CHSH}}^{\mathcal{B}}$, Bob plays the role of the referee as well as the two players and plays TiltedCHSH game.
5. For every i -th sample in $\Gamma_{\text{CHSH}}^{\mathcal{B}}$, Bob randomly generates input bits x_i and y_i for his two measurement devices (these devices act as separate parties without any communication), with $x_i, y_i \in \{0, 1\}$.
6. Bob performs $\text{TiltedCHSH}(\Gamma_{\text{CHSH}}^{\mathcal{B}}, \text{Bob})$, according to the procedure outlined in algorithm 6 for the set $\Gamma_{\text{CHSH}}^{\mathcal{B}}$.
7. If Bob passes this $\text{TiltedCHSH}(\Gamma_{\text{CHSH}}^{\mathcal{B}}, \text{Bob})$ test then both Alice and Bob proceed further, otherwise they abort.
8. From the rest $(\mathcal{K} - \frac{\gamma_1 \mathcal{K}}{2})$ shared states, Alice randomly chooses $\frac{\gamma_1 \mathcal{K}}{2}$ instances, declares those instances publicly and constructs the set $\Gamma_{\text{CHSH}}^{\mathcal{A}}$ with these chosen instances.
9. For all the instances in $\Gamma_{\text{CHSH}}^{\mathcal{A}}$, Bob sends his qubits to Alice.
10. For these instances in $\Gamma_{\text{CHSH}}^{\mathcal{A}}$, Alice plays the role of the referee as well as the two players and plays TiltedCHSH game.

Algorithm 6: TiltedCHSH(\mathcal{S}, \mathcal{P})

- For every $i \in \mathcal{S}$, \mathcal{P} does the following.
 1. If $y_i = 0$, \mathcal{P} 's device applies the measurement operator B_0^0 or B_1^0 randomly on the i -th state's first qubit and generates the output bits $b_i = 0$ and $b_i = 1$ respectively.
 2. If $y_i = 1$, \mathcal{P} 's device applies the measurement operator B_0^1 or B_1^1 randomly on the i -th state's first qubit and generates the output bits $b_i = 0$ and $b_i = 1$ respectively.
 3. Similarly, if $x_i = 0$, \mathcal{P} 's device applies the measurement operator $A_0'^0$ or $A_1'^0$ randomly on the i -th state's second qubit and generates the output bits $a_i = 0$ and $a_i = 1$ respectively.
 4. If $x_i = 1$, \mathcal{P} 's device applies the measurement operator $A_0'^1$ or $A_1'^1$ randomly on the i -th state's second qubit and generates the output bits $a_i = 0$ and $a_i = 1$ respectively.
- From these inputs and outputs, the following quantity is estimated by \mathcal{P} .

$$\begin{aligned} \beta_{\mathcal{B}} &= \alpha_{\mathcal{B}} \sum_{a \in \{0,1\}} (-1)^a \langle \psi_{\mathcal{B}\mathcal{A}} | \mathbb{I} \otimes A_a'^0 | \psi_{\mathcal{B}\mathcal{A}} \rangle \\ &+ \sum_{x,y,a,b \in \{0,1\}} (-1)^{d_{xyab}} \langle \psi_{\mathcal{B}\mathcal{A}} | B_b^y \otimes A_a'^x | \psi_{\mathcal{B}\mathcal{A}} \rangle, \end{aligned}$$

where $\alpha_{\mathcal{B}} = \frac{2}{\sqrt{1+2\tan^2\theta}}$ (for the same θ chosen for the states) and d_{xyab} is defined as follows,

$$d_{xyab} := \begin{cases} 0 & \text{If } xy = a \oplus b \\ 1 & \text{otherwise.} \end{cases}$$

- If $\beta_{\mathcal{B}} = \frac{4}{\sqrt{1+\sin^2\theta}}$ (for the θ chosen for the states) then \mathcal{P} continues with the protocol, otherwise \mathcal{P} aborts the protocol.

11. For every i -th sample in Γ_{CHSH}^A , Alice randomly generates input bits x_i and y_i for her two measurement devices (these devices act as separate parties without any communication), with $x_i, y_i \in \{0, 1\}$.
12. Alice performs $\text{TiltedCHSH}(\Gamma_{\text{CHSH}}^A, \text{Alice})$, according to the procedure outlined in algorithm 6 for the set Γ_{CHSH}^A .
13. If Alice passes the $\text{TiltedCHSH}(\Gamma_{\text{CHSH}}^A, \text{Alice})$ test then both Alice and Bob proceed to the next phase where Alice self-tests her measurement device.

Algorithm 7: OBStestAlice(\mathcal{S})

- Bob has already measured his share of every i -th state of the remaining instances for inputs $y_i = 0$ and $y_i = 1$, and obtained outputs $b_i = 0$ or $b_i = 1$.
- Similarly, Alice has already measured her share of every i -th state of the remaining instances for inputs $x_i = 0$ and $x_i = 1$, obtained outputs $a_i = 0$ or $a_i = 1$, and sent the commitments of those a_i values to Bob.
- For every $i \in \mathcal{S}$, Bob and Alice do the following-

1. Alice reveals the commitments of a_i values only for the instances chosen in the set \mathcal{S} .
2. Bob then estimates the following quantity from the declared outcomes,

$$\beta_{\mathcal{A}} = \frac{1}{4} \sum_{x,y,a,b \in \{0,1\}} (-1)^{d'_{xyab}} \alpha_{\mathcal{A}}^{1 \oplus y} \langle \psi_{\mathcal{BA}} | B_b^y \otimes A_a^x | \psi_{\mathcal{BA}} \rangle$$

where $\alpha_{\mathcal{A}} = \cot \theta$ (for the same θ chosen for the shared states) and d'_{xyab} is as follows,

$$d'_{xyab} := \begin{cases} 0 & \text{If } xy = a \oplus b \\ 1 & \text{otherwise.} \end{cases}$$

3. If $\beta_{\mathcal{A}} = \frac{1}{2 \sin \theta}$ (for the θ chosen for the shared states) then Bob continues with the protocol, otherwise Bob aborts the protocol.

Alice's Measurement Device Verification Phase:

1. Alice and Bob consider the rest $(\mathcal{K} - \gamma_1 \mathcal{K})$ states and do the following.
 - For every i -th state, Bob randomly generates an input bit $x_i \in_R 0, 1$ for Alice's device and publicly declares all $(\mathcal{K} - \gamma_1 \mathcal{K})$ x_i values. After all x_i values are declared, Alice acknowledges receipt to Bob.

- Bob further generates another random bit $y_i \in_R \{0, 1\}$ for every i -th state, as the input of his device.
 - If $y_i = 0$, Bob applies measurement operator B_0^0 or B_1^0 randomly on his share of the i -th state and generates the output bit $b_i = 0$ and $b_i = 1$ respectively.
 - If $y_i = 1$, Bob applies measurement operator B_0^1 or B_1^1 randomly (here $B_0^1 = B_0^0$ and $B_1^1 = B_1^0$) on his share of the i -th state and generates the output bit $b_i = 0$ and $b_i = 1$ respectively.
 - Similarly, if $x_i = 0$, Alice applies measurement operator A_0^0 or A_1^0 randomly on her share of the i -th state and generates the output bit $a_i = 0$ and $a_i = 1$ respectively.
 - If $x_i = 1$, Alice applies measurement operator A_0^1 or A_1^1 randomly on her share of the i -th state and generates the output bit $a_i = 0$ and $a_i = 1$ respectively.
 - Alice encodes all her a_i values using a *computational hiding perfect binding* commitment scheme (Computationally hiding statistically binding commitment schemes are easy to design from a pseudo-random generator and one-way permutation [84, 1, 2]. As these schemes are perfectly binding, Alice can't cheat at all. For the hiding part, we assume that Bob has limitations on his computational resources and he is a polynomial adversary. That means, we assume that Bob can try at most polynomial time to guess a committed bit value. In the multi-client scenario, it is also possible to use some relativistic bit commitment schemes [76, 35, 48]. However, these are outside the scope of this work.) and send those commitments of a_i values to Bob (The inclusion of a commitment scheme is crucial in this context because here Alice performs a non-optimal projective measurement at her end. This introduces the possibility that she might perform the exact projective measurement during the testing phases and later switch to the optimal POVM measurement discussed in Section 5.3 for the instances used in the private query phase. To eliminate this possibility, bit commitment is required as it prevents Alice from postponing measurements for any of her particles and ensures that Alice measures all her particles using the actual projective measurement).
2. Bob then chooses $\gamma_2(\mathcal{K} - \gamma_1\mathcal{K})$ instances randomly from these rest $(\mathcal{K} - \gamma_1\mathcal{K})$ instances, constructs a set Γ_{obs} with those chosen instances and declares those instances publicly.
 3. Alice reveals the commitments of a_i values for all the instances in Γ_{obs} .
 4. Bob then performs $\text{OBStestAlice}(\Gamma_{\text{obs}})$, by following the procedure mentioned in algorithm 7, for the set Γ_{obs} .
 5. If Alice passes the $\text{OBStestAlice}(\Gamma_{\text{obs}})$ then Bob and Alice proceed to the next phase of the protocol where they generate the raw key bits at their end.

Key Generation Phase:

- Alice and Bob consider the rest $(\mathcal{K} - |\Gamma_{\text{CHSH}}| - |\Gamma_{\text{obs}}|)$ samples and construct a set Γ_{QPQ} with those instances where $|\Gamma_{\text{QPQ}}| = kN$.
- For $1 \leq i \leq (|\Gamma_{\text{QPQ}}|)$, Bob and Alice do the following.
 - If Bob’s measurement device generates the outcome $b_i = 0(b_i = 1)$ for the i -th shared state, Bob considers $R_i = 0(R_i = 1)$.
 - Alice already knows the a_i values for all these instances. If Alice’s measurement device receives the input $x_i = 1(x_i = 0)$ and generates the outcome $a_i = 1$ for her share of the i -th state, Alice considers $R_{\mathcal{A}_i} = 0(R_{\mathcal{A}_i} = 1)$.
 - If Alice’s measurement device receives the input $x_i = 0$ or $x_i = 1$ and generates the outcome $a_i = 0$ for her share of the i -th state, Alice remains inconclusive about the value of the raw key bit indexed by i .

Private Query Phase:

Alice and Bob perform the following steps (as mentioned in [117]) for the rest $|\Gamma_{\text{QPQ}}|$ samples.

- Alice and Bob share a kN bit raw key after the *shared key generation phase*, with Bob having full knowledge of the raw key and Alice knowing some unknown bits (corresponding indices unknown to Bob).
- The raw key is divided into k substrings of length N and a bitwise XOR operation is performed to produce the N bit final key.
- If Alice wants to retrieve the bit indexed by j of the database and knows only the i -th bit F_i of Bob’s final key F , she declares the shift number $s = (i - j)$.
- Bob shifts his key F by s positions and encrypts the database using one-time pad.
- The encrypted database can be retrieved by Alice as the j -th database bit is encrypted with F_i (the final key bit indexed by i) known to her.

A schematic diagram of this full DI proposal for the QPQ scheme [117] is shown in the right subfigure (i.e., Subfigure (b)) of Figure 5-3.

5.4.2 Analysis of our scheme

Here, we examine the performance of the proposed full DI version of the QPQ scheme [117]. First, we determine the values of relevant parameters. Then, we evaluate the DI security of our proposed scheme. Finally, we assess the security of the database and user in our proposal.

Note that here we present all our analyses considering the asymptotic scenario. In reality, the values of different parameters (derived here) may deviate from their derived value depending on the chosen sample size.

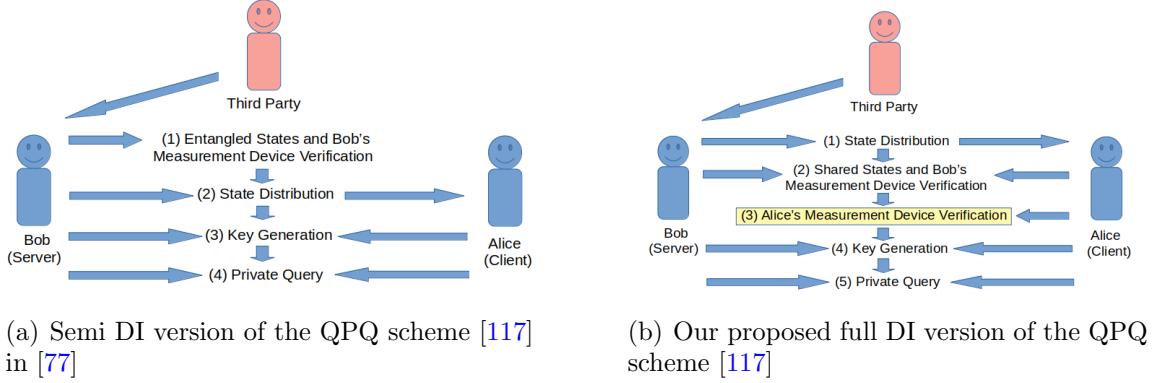


Figure 5-3: Schematic diagram of the semi DI-QPQ scheme [77] (left) and our proposed full DI version of [117] (right)

Estimation of parameters for private query phase

Here, we perform parameter estimation for maintaining both user and data privacy. In this scheme, after the *shared key generation phase*, Bob has kN many raw key bits such that Bob knows all the bits but Alice knows only some of those bits. In the *private query phase*, both Bob and Alice cut their raw keys in some particular positions to prepare N substrings of length k such that $k = \frac{|\Gamma_{\text{QPQ}}|}{N}$ where $|\Gamma_{\text{QPQ}}|$ denotes the total number of samples at the *private query phase* and N denotes the number of database bits.

Estimation of θ for improved security :

Similar to the QPQ scheme [15], here also the server Bob wants the client Alice to retrieve only one data bit in a single query for database security.

In [117], Alice and Bob share kN raw key bits, with Alice able to retrieve on average $\left(\frac{\sin^2 \theta}{2}\right)$ fraction of them. The expected number of raw key bits known to Alice after the *shared key generation phase* (denoted as n_r here) can be calculated as follows,

$$E[n_r] \approx \left(\frac{\sin^2 \theta}{2}\right) kN. \quad (5.1)$$

Alice's probability of correctly guessing a final key bit is roughly $P_f \approx \left(\frac{\sin^2 \theta}{2}\right)^k$ since she must correctly guess all k corresponding raw key bits, which are XORed to form the final key bit.

Here, the number of final key bits known by Alice, n_f (let's say), is a binomial random variable with N total bits and a success probability of $P_f = \left(\frac{\sin^2 \theta}{2}\right)^k$. So,

the expected number of final key bits known by Alice after the *shared key generation phase* is,

$$E[n_f] = P_f N \approx \left(\frac{\sin^2 \theta}{2} \right)^k N. \quad (5.2)$$

In the scheme, dishonest Alice needs to perform correct basis measurements (as specified in [117]) to successfully complete DI testing phases. That means, if the protocol does not abort, the maximum probability of dishonest Alice in guessing R_i (the raw key bit indexed by i) correctly will be atmost $\frac{\sin^2 \theta}{2}$ i.e.,

$$\Pr[R_{\mathcal{A}_i^*} = R_i] \leq \frac{\sin^2 \theta}{2}, \quad (5.3)$$

where \mathcal{A}_i^* denotes dishonest Alice's subsystem corresponding to the i -th shared state.

It is clear that after Bob's measurement, Alice's states are independent and we assume that the measurement devices at dishonest Alice's side are also independent and memoryless. So, the guessing probability of dishonest Alice for F_i (i.e., the final key bit indexed by i) will be upper bounded by $\left(\frac{\sin^2 \theta}{2} \right)^k$ i.e.,

$$\Pr[F_{\mathcal{A}_i^*} = F_i] = P_f \leq \left(\frac{\sin^2 \theta}{2} \right)^k. \quad (5.4)$$

Based on the equations 5.2 and 5.4, it can be seen that the maximum expected number of final key bits that a dishonest Alice can correctly guess, assuming the protocol does not abort, will be limited to a maximum of,

$$E[F_{\mathcal{A}^*}] \leq \left(\frac{\sin^2 \theta}{2} \right)^k N. \quad (5.5)$$

In our scheme, the expected number of data bits correctly guessed by dishonest Alice in a single query is also limited to $\left(\frac{\sin^2 \theta}{2} \right)^k N$ as the database is encrypted by XORing with the final key and correctly guessing a final key bit implies correctly guessing a corresponding database bit, provided the protocol does not abort. This implies that,

$$E[D_{\mathcal{A}^*}] \leq \left(\frac{\sin^2 \theta}{2} \right)^k N. \quad (5.6)$$

In our scheme, for the protocol to continue, Alice must know at least one final key bit, while Bob wants Alice to know less than two final key bits. Thus, the following

condition must be met in the non-abort scenario.

$$1 \leq E[n_f] < 2.$$

This implies that,

$$\begin{aligned} 1 &\leq \left(\frac{\sin^2 \theta}{2}\right)^k N < 2 \\ \frac{1}{N} &\leq \left(\frac{\sin^2 \theta}{2}\right)^k < \frac{2}{N}. \end{aligned} \tag{5.7}$$

All these results boil down to the following conclusion.

Corollary 6. *To ensure that the client Alice only knows less than two final key bits and the proposal doesn't terminate, the server Bob must select the values of θ and the parameter k such that,*

$$\frac{1}{N} \leq \left(\frac{\sin^2 \theta}{2}\right)^k < \frac{2}{N}.$$

Estimation of P_a and P_c for improved security :

Here, we first determine the likelihood that the protocol will not terminate in an honest scenario. Then using the derived bound on the value of $\sin^2 \theta$, we can obtain a lower bound on the value of P_c from the Chernoff-Hoeffding inequality [59] (we estimate the value of P_c using Chernoff-Hoeffding inequality because we consider here the *i.i.d.* scenario).

In our proposal, the likelihood of Alice not correctly guessing a final key bit is calculated as $\left[1 - \left(\frac{\sin^2 \theta}{2}\right)^k\right]$ based on the success probability of Alice in guessing a final key bit, which is $\left(\frac{\sin^2 \theta}{2}\right)^k$.

So, the probability that Alice does not know any of the N final key bits is approximately,

$$\left[1 - \left(\frac{\sin^2 \theta}{2}\right)^k\right]^N \approx e^{-\left(\frac{\sin^2 \theta}{2}\right)^k N}. \tag{5.8}$$

That means the following bound on P_a can be obtained for our proposed scheme.

$$P_a \leq e^{-\left(\frac{\sin^2 \theta}{2}\right)^k N}. \tag{5.9}$$

If Bob sets θ such that $\left(\frac{\sin^2\theta}{2}\right)^k = \frac{1}{N}$, then equation 5.9 gives us the following result according to the relation in equation 5.7.

$$\boxed{P_a \leq e^{-1}}. \quad (5.10)$$

That means this scheme offers a small P_a value. So, the likelihood of the proposal not aborting in the honest scenario (i.e., Alice knowing at least one final key bit) is

$$\begin{aligned} & \Pr(\text{scheme doesn't terminate in honest scenario}) \\ & \geq [1 - e^{-1}]. \end{aligned} \quad (5.11)$$

So, our proposed scheme has a high probability of not aborting in the honest scenario. We now refer to the Chernoff-Hoeffding inequality [59] which is already mentioned in Chapter 4 Proposition 1.

To derive the bound on P_c , we consider $X_i = 1$ when Alice knows the value of the final key bit indexed by i (or its corresponding data bit) in a non-abort scenario (meaning all raw key bits related to the final key bit indexed by i give either $|\phi_0^\perp\rangle$ or $|\phi_1^\perp\rangle$ as an outcome at Alice's side). If the final key has N many bits, the random variable X is defined as $X = \sum_{i=1}^N X_i$.

We have already determined that in the scenario where the proposal doesn't terminate, the expected final key bits that Alice knows is $Y = \left(\frac{\sin^2\theta}{2}\right)^k N$ out of a total of N final key bits. To ensure that the number of known final key bits (X) falls within an error margin $\delta_t = \epsilon \left(\frac{\sin^2\theta}{2}\right)^k N$ (where ϵ is a small constant that depends on the number of samples, one may refer to Section 4.4 of Chapter 4 for details), we use the Chernoff-Hoeffding inequality. This is because the final key bits are independent and the measurement devices at Alice's end are also independent and memoryless. The calculations of X and Y are based on the non-abort scenario. So, we can write the following from the Chernoff-Hoeffding inequality in Proposition 1.

$$\begin{aligned} & \Pr[|X - Y| < \delta_t \wedge \text{scheme doesn't terminate}] \\ & \geq 1 - \exp(-2\delta_t^2 m). \end{aligned} \quad (5.12)$$

After the *shared key generation phase*, Bob has N final key bits and we want Alice's known final key bits to fall within the range of $[p - \epsilon p, p + \epsilon p]$, where $p = \left(\frac{\sin^2\theta}{2}\right)^k N$ and $\delta_t = \epsilon \left(\frac{\sin^2\theta}{2}\right)^k N$ is the accepted deviation. Plugging in δ_t and $m = N$ into

equation 5.12 gives,

$$\boxed{\begin{aligned} &\Pr [|X - Y| < \delta_t \wedge \text{scheme doesn't terminate}] \\ &\geq 1 - \exp(-2\delta_t^2 N) \\ &\text{where } \delta_t = \epsilon \left(\frac{\sin^2 \theta}{2} \right)^k N. \end{aligned}} \quad (5.13)$$

In equation 5.7, the following bound is already derived on $\left(\frac{\sin^2 \theta}{2} \right)^k$.

$$\frac{1}{N} \leq \left(\frac{\sin^2 \theta}{2} \right)^k < \frac{2}{N}.$$

By setting $\left(\frac{\sin^2 \theta}{2} \right)^k = \frac{1}{N}$ in equation 5.13, we obtain,

$$\boxed{\begin{aligned} &\Pr [|X - Y| < \epsilon \wedge \text{scheme doesn't terminate}] \\ &\geq 1 - \exp(-2\epsilon^2 N). \end{aligned}}$$

If the scheme is implemented honestly, the following lower bound of the parameter P_c can be obtained from the definition 1 as guessing a final key bit correctly means correctly guessing the corresponding data bit.

$$\boxed{P_c \geq [1 - \exp(-2\epsilon^2 N)].} \quad (5.14)$$

As in practice, N is large, this probability will be significant. That means, in case of honest implementation of our proposed scheme, the probability that Alice knows the expected number of data bits (with atmost ϵ deviation from the expected number) and the scheme does not terminate is high.

The bound on δ_t can be obtained from equation 5.7 as $\delta_t = \epsilon \left(\frac{\sin^2 \theta}{2} \right)^k N$.

$$\epsilon \leq \delta_t < 2\epsilon. \quad (5.15)$$

From this, it's clear that ϵ must satisfy the constraint $2\epsilon \leq 1$, resulting in an upper bound of $\epsilon \leq \frac{1}{2}$. Now we move to discuss the security concerns of our proposal.

Security in device independent scenario

In this work, we propose a full DI version of the QPQ scheme [117]. The correctness of this scheme is already mentioned in [77]. Hence, we mention here only the security related issues of our full DI proposal. Based on the results obtained from Theorem 8 and Theorem 9, here we conclude about the DI security of the QPQ scheme [117].

Theorem 8. *(DI testing of shared states and Bob's measurement devices) In the TiltedCHSH test of the source device and Bob's measurement device verification phase, either the devices achieve $\beta_B = \frac{4}{\sqrt{1+\sin^2 \theta}}$ for both Alice and Bob (i.e., the states*

provided by the third party are identical with the actual states as mentioned in the QPQ scheme [117] and Bob's measurement device measures correctly in the $\{|0\rangle, |1\rangle\}$ basis) or the scheme is likely to abort with high probability (as the limit approaches infinity).

The proof of this theorem exactly follows from the results mentioned in [8] and [14]. We present an outline of this proof later in Section 5.6.

So, Theorem 8 guarantees that either the states shared between Alice and Bob are of the specified form and Bob's measurement device measures correctly in $\{|0\rangle, |1\rangle\}$ basis or the scheme terminates with high likelihood (as the limit approaches infinity). The next DI testing is done in *Alice's measurement device verification phase*. This phase basically guarantees the functionality of Alice's measurement device. Alice and Bob lead to this phase whenever they successfully pass the first DI testing phase. In this phase, Alice measures in $\{|\phi_0\rangle, |\phi_0^\perp\rangle\}$ or $\{|\phi_1\rangle, |\phi_1^\perp\rangle\}$ basis randomly whereas Bob measures in $\{|0\rangle, |1\rangle\}$ basis. From the measurement outcome, they estimate the value of a parameter β_A and check whether this value is equal to $\frac{1}{2\sin\theta}$. Theorem 9 guarantees that either Alice's devices measure correctly in the specified basis, resulting in $\beta_A = \frac{1}{2\sin\theta}$, or the protocol will abort with high probability as the limit approaches infinity.

Theorem 9 (DI testing of Alice's measurement devices). *In OBStestAlice, either Alice's measurement devices achieve the value of the parameter $\beta_A = \frac{1}{2\sin\theta}$ (i.e., her devices correctly measure in $\{|\phi_0\rangle, |\phi_0^\perp\rangle\}$ and $\{|\phi_1\rangle, |\phi_1^\perp\rangle\}$ basis) or the protocol terminates with a high likelihood of failure (as the limit approaches infinity).*

The proof of this theorem is explained later in detail in Section 5.7 and follows the same method outlined in [65] for certifying non-maximally incompatible observables.

Note: Here, we claim that if Alice and Bob successfully pass both the Tilted-CHSH test and the OBStestAlice mentioned in our full DI proposal, then in the QPQ scheme [117], neither of Alice and Bob can retrieve any additional information in the noiseless scenario. Now, suppose that our claim is wrong i.e., Alice and Bob can pass all the tests mentioned in our scheme and later Alice can retrieve more data bits (than what she intends to know) in a single query or Bob can guess the query indices of Alice with a more certain probability (than his intended probability).

Similar to the analysis in [15], here also we discuss this issue in the context of a particular form of *non-i.i.d.* attack, where a specific number of states are independently corrupted (more general attacks are also possible but these are outside the scope of this work). In this context, we will show that if some of the corrupted states are included during the testing phases, then there is some probability of being caught as the limit approaches infinity.

At the beginning of our scheme, the untrusted third party provides all the states to the server Bob and then Bob shares those states with Alice. As in the *source device and Bob's measurement device verification phase*, both the parties choose the states randomly from the shared instances for the local tests at their end, the dishonest party can not guess beforehand the shared instances that the honest party will choose at

his end for the local test. According to our assumption, the dishonest party can not manipulate the honest party's device once the protocol starts. So, to successfully pass the TiltedCHSH test at the honest party's end, the shared states must be of the actual form as specified in [117]. Similarly, in the TiltedCHSH test performed at Bob's side, the honest Bob must measure the states in the specified basis (to detect the corrupted states) which also certifies the specific measurement bases of Bob. This implies that the *source device and Bob's measurement device verification phase* certifies all the states provided by the untrusted third party and also certifies the measurement device of Bob for the standard basis.

We now explain these things more formally. Let us suppose that initially, the untrusted third party colludes with either the dishonest Alice or the dishonest Bob and shares either \mathcal{K}_A corrupted states in favour of Alice (let us denote this type of states as \mathcal{A} -type) or \mathcal{K}_B corrupted states in favour of Bob (let us denote this type of states as \mathcal{B} -type) among \mathcal{K} shared states. So, while choosing randomly for the TiltedCHSH test at honest Bob's end, the probability that a chosen state is of \mathcal{A} -type is $\frac{\mathcal{K}_A}{\mathcal{K}}$. Similarly, for the TiltedCHSH test at honest Alice's end, the probability that a chosen state is of \mathcal{B} -type is $\frac{\mathcal{K}_B}{\mathcal{K}}$. Let us further assume that for the \mathcal{A} -type states, the value of the parameter β_B is β'_A (where $\beta'_A = \beta_B + \epsilon_A$ such that $\epsilon_A > 0$) and for the \mathcal{B} -type states, the value of the parameter β_B is β'_B (where $\beta'_B = \beta_B + \epsilon_B$ such that $\epsilon_B > 0$).

Now, suppose that only Alice is dishonest and the third party supplies \mathcal{K}_A number of corrupted states (in favour of Alice) along with $(\mathcal{K} - \mathcal{K}_A)$ actual states. Then, in the local test at Bob's end, the probability that a chosen state is not of the \mathcal{A} -type is $(1 - \frac{\mathcal{K}_A}{\mathcal{K}})$. One can easily check that this probability is also same for a chosen state in the final QPQ phase. As, dishonest Alice's aim is to gain as much additional data bits as possible, she needs to choose the value of \mathcal{K}_A such that $(\mathcal{K} - \mathcal{K}_A) = c$ where c is exponentially smaller than \mathcal{K} (i.e., she will try to maximize the probability that a state chosen for the final QPQ phase is of the \mathcal{A} type). Then, the probability that Bob will choose none of the corrupted states (i.e., the \mathcal{A} type states) among his chosen $\frac{\gamma_1 \mathcal{K}}{2}$ states for the TiltedCHSH test at his end is,

$$\left(1 - \frac{\mathcal{K}_A}{\mathcal{K}}\right)^{\frac{\gamma_1 \mathcal{K}}{2}} = \left(\frac{c}{\mathcal{K}}\right)^{\frac{\gamma_1 \mathcal{K}}{2}},$$

which is negligible in \mathcal{K} . Similarly, whenever Bob is dishonest, the same thing can be shown for the local TiltedCHSH test at Alice's end. This implies that if the third party colludes with the dishonest party and supplies corrupted states then the probability that none of those corrupted states will be chosen for the local test at the honest party's end is negligible.

In our scheme, we consider the ideal scenario where there is no channel noise. So for dishonest Alice, to successfully pass the TiltedCHSH test at the honest Bob's end, the following relation must hold in the noiseless condition.

$$\begin{aligned}
\frac{\mathcal{K}_A \beta'_A}{\mathcal{K}} + \frac{(\mathcal{K} - \mathcal{K}_A) \beta_B}{\mathcal{K}} &= \beta_B \\
\mathcal{K}_A \beta'_A + (\mathcal{K} - \mathcal{K}_A) \beta_B &= \mathcal{K} \beta_B \\
\mathcal{K}_A (\beta'_A - \beta_B) &= 0.
\end{aligned}$$

Now, replacing the values of β'_A from the relation $\beta'_A = \beta_B + \epsilon_A$, one can get,

$$\mathcal{K}_A \epsilon_A = 0. \tag{5.16}$$

As the value of $\epsilon_A > 0$, from this relation, one can easily conclude that in the noiseless scenario, the value of \mathcal{K}_A must be zero to successfully pass the local test at the honest Bob's end. Similarly, one can show that whenever Bob is dishonest, the value of \mathcal{K}_B must be zero to successfully pass the local test at the honest Alice's end. In practice, for finite number of samples, one can show that the values of \mathcal{K}_A and \mathcal{K}_B must be negligible to successfully pass the local test at the honest party's end.

In this proposal, we consider a scenario where the shared states are exchanged between the two parties before the start of the protocol, and the dishonest party cannot manipulate the honest party's device after the start of the protocol. As we focus on the *i.i.d.* case, it's clear from the proof of Theorem 8 in Section 5.6 that either the scheme terminates with high likelihood (as the limit approaches infinity), or the TiltedCHSH test will certify that the shared states in the QPQ scheme [117] reach the desired value of the parameter β_B .

Similarly, the TiltedCHSH test at the honest Bob's end also confirms that either Bob aborts the scheme with high probability (as the limit approaches infinity), or the TiltedCHSH test at his end certifies that his measurement devices achieve the intended value of the parameter β_B .

The next DI testing is done in *Alice's POVM device verification phase* where Bob and Alice perform distributed test to certify Alice's projective measurement device. Here, one may think that if Bob is dishonest, then for the instances chosen in *Alice's POVM device verification phase*, he will measure in the actual measurement basis at his end to detect the fraudulent behaviour of Alice, and later for the instances to be used for the actual QPQ phase, he will measure in some different basis to guess the positions of Alice's known key bits.

From the result obtained in Lemma 4, it is clear that for Bob to guess Alice's query indices with more certainty, he must reveal more data bits to dishonest Alice in a single query. But doing so violates assumption 4, which states that neither Alice nor Bob leaks more information (from their side) to gain additional knowledge from the other party. Therefore, Bob should act honestly for all the instances in *Alice's POVM device verification phase* as well as in *shared key generation phase* to ensure the validity of Alice's measurement device, prevent dishonest Alice from obtaining any additional information, and also to maintain his reputation as a database owner (For our proposal, Bob has a chance to cheat because of the inclusion of the compu-

tational hiding perfect binding commitment scheme. However, we assume that Bob has limitations on his computational resources and he is a polynomial-time adversary. This assumption bounds Bob to guess a committed bit of Alice. It is also impractical that Bob spends more than the polynomial time to retrieve a particular raw key bit. That’s why the computational hiding commitment scheme introduced in our scheme will not leak any additional information to Bob).

As Bob acts honestly for *Alice’s POVM device verification phase* and chooses the input bits randomly for both the parties in OBStestAlice, there is no possibility that the inputs for OBStestAlice are chosen according to some dishonest distribution. As the focus of this proposal is on the *i.i.d.* scenario, it can be easily concluded (based on the proof of Theorem 9 in Section 5.7) that either Alice and Bob will abort the scheme with high likelihood (as the limit approaches infinity), or OBStestAlice will confirm that Alice’s measurement devices achieve the intended value of β_A .

That means we can conclude the following from all these discussions.

Corollary 7. *Our DI scheme either terminates with high likelihood (as the limit approaches infinity) or certifies that the devices in the QPQ scheme [117] achieve the desired values of β_B and β_A in the TiltedCHSH test and OBStestAlice respectively.*

Given the discussion above on some types of *non-i.i.d.* attack in our DI proposal, the statement in corollary 7 can probably be generalized to some *non-i.i.d.* cases, but it is outside the scope of this work.

Security of database against dishonest Alice

Here we estimate the amount of raw key bits that dishonest Alice can guess in the *shared key generation phase*, and the probability of her retrieving more than the expected data bits in a single query. Dishonest Alice can guess additional raw key bits either from the loophole of the underlying bit commitment scheme or by manipulating the other devices and using an optimal measurement device at her side.

For the underlying computational hiding and perfect binding bit commitment scheme using a pseudo-random generator, the security of the database against dishonest Alice follows from Claim 3.1 in [84] which states that for any i -th committed bit a_i , Alice can fool Bob (i.e., Alice can successfully verify the commitment for a different bit other than the committed one) with probability at most 2^{-n} where n is the security parameter which is chosen such that no feasible machine can break the underlying pseudorandom generator for seeds of length n . That means, dishonest Alice can’t retrieve more raw key bits and if she tries to do so and commits the a_i values obtained from the optimal measurement then it will be detected by Bob during OBStestAlice. More precisely, the security of the entire bit commitment protocol follows from the result mentioned in [84, Theorem 3.1] which states the following.

Corollary 8. *If the underlying device G is a pseudorandom generator, then for all polynomials p and large enough security parameter n , the corresponding bit commitment protocol obeys the following.*

- After commitment, no probabilistic polynomial-time Bob can guess any committed a_i value with probability greater than $\left(\frac{1}{2} + \frac{1}{p(n)}\right)$.
- Alice can reveal only the committed bit, except with probability less than 2^{-n} .

In the case of the manipulation of the other devices and her device, the estimation follows from the DI results in corollary 7 which states that after the DI testing phases, either the scheme will abort with high probability (as the limit approaches to infinity) or the devices involved in [117] will meet the intended values of parameters β_A and β_B as indicated in our proposal.

Theorem 10. *In our scheme, in the absence of $OBStestAlice$, dishonest Alice can inconclusively retrieve (i.e., the indices of the correctly guessed bits are unknown) $\left(\frac{1}{2} + \frac{1}{2} \sin \theta\right)$ fraction of the entire raw key during the shared key generation phase.*

The proof of this Theorem directly follows from the proof of Theorem 5 in [15] (i.e., the proposal mentioned in Chapter 4). The only difference here is that in this scheme, Alice needs to distinguish between the two non orthogonal quantum states $|\phi_0\rangle$ and $|\phi_1\rangle$ as compared to the two non orthogonal states $|0\rangle$ and $|0'\rangle$ (or $|1\rangle$ and $|1'\rangle$) in [15] (i.e., the proposal mentioned in Chapter 4).

In our full DI proposal, dishonest Alice (\mathcal{A}^*) can not perform any other measurement other than the projective measurement mentioned in [117] because if she performs any other measurement at her side then it will be detected in *Alice's POVM device verification phase*. Because of this, we can get a bound on the number of raw key bits that dishonest Alice can retrieve (on average) in this full DI proposal of the QPQ scheme [117].

Lemma 3. *Either our protocol terminates with high likelihood in the long run, or dishonest Alice (\mathcal{A}^*) can retrieve (on average) $\frac{\sin^2 \theta}{2}$ fraction of bits from the entire raw key after the shared key generation phase.*

Proof. According to the QPQ scheme [117], after the measurements at the server Bob's side, the client Alice has kN independent non-orthogonal qubits at her end. For each of the instances, Alice now tries to distinguish between the non-orthogonal states $|\phi_0\rangle$ and $|\phi_1\rangle$.

From the QPQ scheme [117], it is clear that if Alice measures her qubits in $\{|\phi_0\rangle, |\phi_0^\perp\rangle\}$ and $\{|\phi_1\rangle, |\phi_1^\perp\rangle\}$ basis randomly, then Alice can guess a raw key bit with certainty whenever the outcome is either $|\phi_0^\perp\rangle$ or $|\phi_1^\perp\rangle$.

From the correctness of the QPQ scheme [117], it is clear that for each of the instances, the probability of getting the outcome $|\phi_0^\perp\rangle$ or $|\phi_1^\perp\rangle$ using projective measurement is $\frac{\sin^2 \theta}{2}$.

Our DI proposal requires dishonest Alice to independently measure each of the kN qubits at her end in a specified basis to pass the testing phases. If she performs random and independent projective measurements in the $|\phi_0\rangle, |\phi_0^\perp\rangle$ and $|\phi_1\rangle, |\phi_1^\perp\rangle$ basis, on average, she can retrieve $\left(\frac{\sin^2 \theta}{2}\right) kN$ raw key bits correctly. This concludes the proof. \square

In this DI proposal, the database contains N data bits. Although there is a chance that dishonest Alice can successfully pass all tests and learn more data bits than allowed through statistical fluctuations, the likelihood of this happening is low according to Corollary 7. Now, based on Definition 3 and equation 5.6, we can conclude the following.

Corollary 9. *In the case of dishonest Alice and honest Bob, either the proposed scheme will likely abort (as the limit approaches infinity) or dishonest Alice will, on average, be able to obtain τ fraction of bits from the entire final key, where*

$$\tau \leq \left(\frac{\sin^2 \theta}{2} \right)^k. \quad (5.17)$$

By using the upper bound from equation 5.7 in place of $\left(\frac{\sin^2 \theta}{2} \right)^k$, we can obtain the following upper limit for the value of τ .

$$\boxed{\tau < \frac{2}{N}}. \quad (5.18)$$

It shows that our full DI proposal results in τ being significantly smaller than N .

It is possible to validate the data privacy of our scheme in another way (other than the data privacy definition mentioned in Definition 3) showing that the probability with which dishonest Alice can successfully guess more than the expected number of final key (or equivalently data) bits (with a deviation more than the ϵ fraction from the expected number) such that the protocol doesn't terminate is low.

Like the discussion in Subsection 5.4.2 (entitled “estimation of parameters for private query phase”), here also we assume that the random variable X denotes the number of final key bits known to the dishonest Alice and Y be the expected value in honest scenario.

Here, we shall prove that the probability $\Pr[|X - Y| > \delta_t \wedge \text{scheme doesn't terminate}]$ is negligible. In general, this can be shown using the properties of basic probability theory. As we consider the *i.i.d.* assumption in our proposal, there will be two different subcases- 1) all the devices attain the ideal TiltedCHSH value, or 2) all the devices do not attain the ideal TiltedCHSH value.

Note that $\Pr[|X - Y| > \delta_t \wedge \text{scheme doesn't terminate}]$ is upper bounded by both $\Pr[|X - Y| > \delta_t]$ and $\Pr[\text{scheme doesn't terminate}]$, according to the property of basic probability theory (which says $\Pr[A \wedge B] \leq \Pr[A]$ and $\Pr[A \wedge B] \leq \Pr[B]$).

Now for the first subcase, from the DI security statement in Theorem 9 (more precisely, from the self-testing argument of Theorem 9), one can easily conclude that $\Pr[|X - Y| > \delta_t] \leq \text{negl}(N)$.

For the second subcase, by an analysis similar to the proof of Theorem 9, it can be concluded that $\Pr[\text{scheme doesn't terminate}] \leq \text{negl}(N)$. This implies that for both of these two subcases, $\Pr[|X - Y| > \delta_t \wedge \text{scheme doesn't terminate}] \leq \text{negl}(N)$ (under the *i.i.d.* assumption).

Security of user against dishonest Bob

In this subsection, we determine the number of indices (l_{B^*}) that dishonest Bob can accurately guess from \mathcal{I}_l (the query index set of Alice). Additionally, we calculate the probability of Bob correctly guessing more indices than expected. Generally, for any QKD-based QPQ schemes, if Bob attempts to cheat, there is a risk of providing false information about the intended data bits to Alice, potentially harming his reputation as a database owner [63]. Therefore, for the QPQ primitive, Bob is assumed not to cheat if there is a non-zero probability of being caught. Our scheme provides Bob a chance to cheat without being detected due to the underlying bit commitment scheme. However, we assume that Bob is a polynomial-time adversary and has computational limitations. For this reason, even with the existence of a computational hiding bit commitment scheme, Bob cannot gain any information about Alice's committed bits. So, the calculation here is only based on the results of corollary 7, which states that either the scheme terminates with high likelihood or the devices in [117] achieve the desired values of β_A and β_B after the DI testing phases. Based on these results and those in [15], we can conclude the following.

Lemma 4. *Dishonest Bob can correctly predict a maximum of $\frac{l}{N}$ fraction of the indices from \mathcal{I}_l after l queries to the N -bit database (in [117]), i.e., for a particular index i ,*

$$\Pr(\text{Bob correctly guesses an index } i \in \mathcal{I}_l) \leq \frac{l}{N}.$$

Proof. At the *shared key generation phase* of our proposal, Alice does not broadcast anything about her measurement outcome. So, dishonest Bob has no information about Alice's measurement outcomes and her known key bits. Now, Alice queries l many times to the database and retrieves l many data bits. After these l many queries, dishonest Bob will try to guess those query indices of Alice. As, Bob has no information about the known final key bits of Alice, he has to guess these l many indices (out of the N data bits) randomly.

So, for any i -th data bit, dishonest Bob can guess whether $i \in \mathcal{I}_l$ with probability atmost $\frac{l}{N}$. This completes the proof. \square

This implies that Bob can guess whether a database index is in \mathcal{I}_l (the query index set of Alice) with a probability of at most $\frac{l}{N}$. Assuming Alice only knows one data bit per query, if Bob guesses l bits, the expected number of correct guesses Bob can make from Alice's query set \mathcal{I}_l will be,

$$\begin{aligned} E[l_{B^*}] &= \Pr(\text{Bob correctly guesses an index } i \in \mathcal{I}_l) \cdot l \\ &\leq \frac{l^2}{N}. \end{aligned} \tag{5.19}$$

This DI-QPQ proposal includes tests to prevent Bob from discovering too much about \mathcal{I}_l (the query index set of Alice), but due to statistical fluctuations, Bob still

has a chance of passing the tests and obtaining more information than a negligible fraction of the indices. As the limit approaches infinity, Bob's likelihood of passing all the tests becomes low according to Corollary 7. Furthermore, if Bob wants to increase the certainty of guessing a query index, he would need to allow Alice to know more data bits (as stated in the result of Lemma 4), which goes against assumption 4.

Comparing the expression in definition 4 with equation 5.19 provides the following upper bound for δ in our proposal.

Corollary 10. *The DI-QPQ proposal will either abort with high likelihood (as the limit approaches infinity), or dishonest Bob will be able to correctly predict, on average, δ fraction of indices from \mathcal{I}_l where,*

$$\delta \leq \left(\frac{l}{N} \right). \quad (5.20)$$

In practice, the number of data bits in the database, N , is significantly larger than the size of \mathcal{I}_l (i.e., l), with N approximately equal to l^n for some positive integer n . Using this information and equation 5.20, the following upper bound on the value of δ can be obtained.

$$\delta \leq \frac{1}{l^{(n-1)}}. \quad (5.21)$$

This equation shows that the value of δ is small compared to l in our proposal.

5.4.3 Comparison with the QPQ scheme mentioned in Chapter 4

Our proposal in Chapter 4 also addresses the same problem of Quantum Private Query in Device Independent scenario. Here we mention a comparative study between the full DI proposal of the QPQ scheme [117] mentioned in this chapter and the full DI-QPQ scheme mentioned in Chapter 4.

- **Total number of samples :**

In the DI-QPQ scheme mentioned in Chapter 4, there are total 6 phases namely entanglement distribution phase, source device verification phase, DI testing phase for Bob's measurement device, DI testing phase for Alice's measurement device, key establishment phase and private query phase. On the other hand, in our proposed DI version of Yang et al. [117] QPQ scheme, there are total 4 phases namely source device and Bob's measurement device verification phase, Alice's measurement device verification phase, key generation phase and private query phase. For consistency and simplicity of comparison, here we consider that each of the protocols starts with N number of samples (i.e., states) and whenever Alice and Bob choose some samples for testing purposes, they choose γ fraction of instances all the time (i.e., for the scheme mentioned in Chapter 4,

here we consider $\gamma_1 = \gamma_2 = \gamma_3 = \gamma$). Here we show that if Alice and Bob start with same number of initial states (i.e., N) for both the protocols and choose γ fraction of samples for all the testing phases, then Alice and Bob can use more number of samples in the private query phase for this proposed full DI version of the QPQ scheme [117] as compared to the number of samples used in private query phase for the DI-QPQ scheme mentioned in Chapter 4.

So, for the DI-QPQ protocol mentioned in Chapter 4, considering $\mathcal{K} = N$ and $\gamma_1 = \gamma_2 = \gamma_3 = \gamma$, Alice and Bob first choose γN samples for their localCHSH test which certifies the given states. Next in OBStest, each of Alice and Bob independently chooses $\frac{\gamma}{2}(N - \gamma N)$ samples randomly from the rest $(N - \gamma N)$ states to certify Bob's measurement device. So, the total number of samples used in the OBStest is $\gamma(N - \gamma N)$. Next in the DI testing phase for Alice's measurement device, Alice chooses γ fraction of samples randomly from the rest $(1 - \gamma)(N - \gamma N)$ samples to certify her measurement device. Atlast, the rest $[(1 - \gamma)(N - \gamma N) - \gamma(1 - \gamma)(N - \gamma N)] = (1 - \gamma)^3 N$ samples are used for private query phase. This implies that in the DI-QPQ scheme mentioned in Chapter 4, the server Bob can generate a raw key of length $(1 - \gamma)^3 N$ bits using N number of states. So, if we consider that in the QPQ scheme mentioned in Chapter 4, Alice and Bob use F_{old} fraction of initial samples for private query phase then $F_{old} = (1 - \gamma)^3$.

Similarly, for the full DI version of the QPQ scheme [117] mentioned here, considering $\mathcal{K} = N$ and $\gamma_1 = \gamma_2 = \gamma$, Bob and Alice first choose γN samples randomly for their local TiltedCHSH test which certifies the given states and Bob's measurement device. Each of Alice and Bob then chooses $\frac{\gamma(N - \gamma N)}{2}$ samples randomly from the rest $(N - \gamma N)$ states for OBStestAlice which certifies Alice's measurement device. Atlast, the rest $(N - \gamma N) - \gamma(N - \gamma N) = (1 - \gamma)^2 N$ samples are used for key generation. This implies that in the full DI version of the QPQ scheme [117] mentioned here, the server Bob can generate $(1 - \gamma)^2 N$ raw key bits using N number of states. So, if we consider that in this scheme, Alice and Bob use F_{new} fraction of initial samples for private query phase then $F_{new} = (1 - \gamma)^2$.

A comparative study between the number of samples used for raw key generation in two different protocols for different values of γ is shown in Figure 5-4. From this figure, it is clear that for any value of γ (where $\gamma \in (0, 1)$), the size of the raw key generated in the proposed full DI version of the QPQ scheme [117] is always greater than the size of the raw key generated in the DI-QPQ scheme mentioned in Chapter 4. This implies that to generate a raw key of a particular size, the DI-QPQ scheme mentioned in Chapter 4 requires more number of initial samples as compared to the full DI version of the QPQ scheme [117] mentioned here. So, in terms of the total number of samples, this full DI version of the QPQ scheme [117] is more efficient as compared to the DI-QPQ scheme mentioned in Chapter 4.

- **Projective measurement Vs. POVM:**

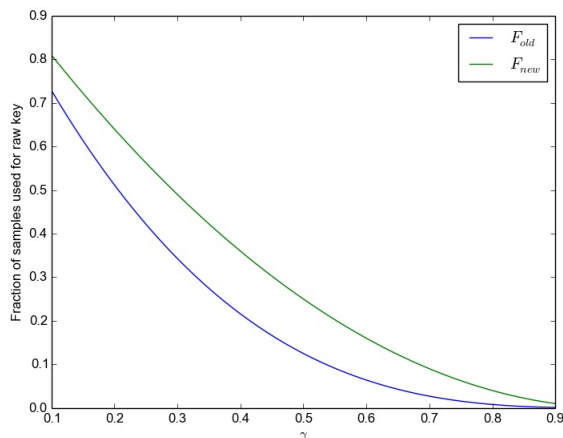


Figure 5-4: Comparison between the fraction of samples used for raw key generation in two different protocols for different values of γ

The DI-QPQ schemes in this chapter and the scheme mentioned in Chapter 4 are QKD-based, using non-orthogonal state distinction for key generation. In the scheme mentioned in Chapter 4, the client uses POVM measurements to distinguish non-orthogonal states, while this chapter’s full DI version of [117] uses projective measurements.

Although it is well-known that POVM measurements can be implemented as a projective measurements in the higher dimension, it requires additional gates as compared to the projective measurements in actual dimension. This implies that the implementation of POVM measurement is complicated as compared to the projective measurement. So, from the viewpoint of practical implementation of the measurement devices, the full DI version of [117] mentioned here is more efficient as compared to the DI-QPQ scheme mentioned in Chapter 4.

- It is well-known that the maximally entangled states are easy to prepare as compared to the non-maximally entangled states. The DI-QPQ scheme mentioned in Chapter 4 uses maximally entangled states whereas the full DI version of the QPQ scheme [117] mentioned here uses non-maximally entangled states. So, from the viewpoint of practical implementation of the source device, the DI-QPQ scheme mentioned in Chapter 4 is more efficient as compared to the DI-QPQ schemes mentioned in this chapter.

5.5 Full DI proposal for a modified version of the QPQ scheme [117]

From the analysis of section 5.3, it is clear that for the QPQ scheme [117], the client Alice can retrieve more number of database bits in a single query, if she performs

optimal POVM measurement at her side instead of the projective measurements mentioned in [117]. In this direction, here we propose a full DI protocol for a modified version of [117] where the client Alice can retrieve optimal number of raw key bits at her end.

We divide this entire section into two subsections. In the first subsection, we propose different steps of our modified DI-QPQ scheme and in the last subsection, we mention the security related issues of this modified proposal. The assumptions for this modified DI-QPQ scheme are also same as the assumptions mentioned in Section 3.6 of Chapter 3.

5.5.1 Modified full DI protocol

Like the previous DI proposal, here also we divide the entire protocol into four phases based on the functionality. The first phase which certifies the state generation device and Bob’s measurement device is termed *Source Device and Bob’s Measurement Device Verification Phase*.

The next phase certifies the measurement devices for the client Alice and is termed *Alice’s Measurement Device Verification Phase*.

After successful completion of these two testing phases, Bob and Alice conclude that the states given to them are of the specified form and their measurement devices measure correctly in the specified bases (here ‘specified’ refers to the state and measurement bases mentioned in this modified QPQ proposal). After these testing phases, Bob and Alice proceed to the *Key Generation Phase* where Bob generates a key and Alice knows some bits of that key such that Bob can not know anything about Alice’s known key bits. At last, they proceed to the *private query phase* where Bob encrypts the entire database using the key generated in the key generation phase and sends it to Alice. Alice then decrypts the intended bits of the database using her partial knowledge about the final key bits.

Now we describe different steps of our entire protocol. Note that like our previous scheme, here also we consider that there is no channel noise i.e., all the operations are perfect.

<p>Algorithm 8: KeyGenAlice(\mathcal{S})</p> <ul style="list-style-type: none"> • For each index $i \in \mathcal{S}$, Alice performs the following steps. <ol style="list-style-type: none"> 1. Alice uses the measurement device $D = \{D_0, D_1, D_2\}$ to measure her qubit of the shared state indexed by i. 2. Alice concludes the raw key bit indexed by i as 0(1) if she gets the measurement outcome $D_0(D_1)$ for the shared state indexed by i. 3. Alice remains uncertain about the raw key bit indexed by i if she gets the measurement outcome D_2 for the shared state indexed by i.

Algorithm 9: POVMtestAlice(\mathcal{S})

- For each index $i \in \mathcal{S}$, Bob and Alice perform the following steps.
 1. Bob first declares the value of R_i (i.e., the raw key bit indexed by i).
 2. Whenever $R_i = 0$ ($R_i = 1$), Alice considers that the state at her side is ρ_0 (ρ_1).

- Alice then computes the parameter

$$\Omega = \sum_{R_i, R_{A_i} \in \{0,1\}} (-1)^{R_i \oplus R_{A_i}} \text{Tr}[D_{R_{A_i}} \rho_{R_i}],$$

where $D_{R_{A_i}}$ is Alice's measurement outcome in KeyGenAlice() for the i -th instance.

- If for the set \mathcal{S} ,

$$\Omega = \frac{2 \sin^2 \theta}{(1 + \cos \theta)},$$

then they continue with the protocol, otherwise they abort.

Source Device and Bob's Measurement Device Verification Phase:

1. Bob starts with \mathcal{K} (we assume here that \mathcal{K} is asymptotically large) number of states (say $|\psi\rangle_{\mathcal{B}\mathcal{A}}$) provided by the third party and shares those states with Alice in such a way that the first particle of each state corresponds to Bob and the second particle corresponds to Alice.
2. Bob chooses $\frac{\gamma\mathcal{K}}{2}$ instances randomly from these \mathcal{K} shared states, declares those instances publicly and constructs the set $\Gamma_{\text{CHSH}}^{\mathcal{B}}$ with these chosen instances.
3. For all the instances in $\Gamma_{\text{CHSH}}^{\mathcal{B}}$, Alice sends her qubits to Bob.
4. For the instances in $\Gamma_{\text{CHSH}}^{\mathcal{B}}$, Bob plays the role of the referee as well as the two players and plays TiltedCHSH game.
5. For every i -th sample in $\Gamma_{\text{CHSH}}^{\mathcal{B}}$, Bob randomly generates input bits x_i and y_i for his two measurement devices (these devices act as separate parties without any communication), with $x_i, y_i \in \{0, 1\}$.
6. Bob performs TiltedCHSH($\Gamma_{\text{CHSH}}^{\mathcal{B}}$, Bob), according to the procedure outlined in algorithm 6 for the set $\Gamma_{\text{CHSH}}^{\mathcal{B}}$.
7. If Bob passes this TiltedCHSH($\Gamma_{\text{CHSH}}^{\mathcal{B}}$, Bob) test then both Alice and Bob proceed further, otherwise they abort.

8. From the rest $(\mathcal{K} - \frac{\gamma_1 \mathcal{K}}{2})$ shared states, Alice randomly chooses $\frac{\gamma_1 \mathcal{K}}{2}$ instances, declares those instances publicly and constructs the set Γ_{CHSH}^A with these chosen instances.
9. For all the instances in Γ_{CHSH}^A , Bob sends his qubits to Alice.
10. For these instances in Γ_{CHSH}^A , Alice plays the role of the referee as well as the two players and plays TiltedCHSH game.
11. For every i -th sample in Γ_{CHSH}^A , Alice randomly generates input bits x_i and y_i for her two measurement devices (these devices act as separate parties without any communication), with $x_i, y_i \in \{0, 1\}$.
12. Alice performs $\text{TiltedCHSH}(\Gamma_{\text{CHSH}}^A, \text{Alice})$, according to the procedure outlined in algorithm 6 for the set Γ_{CHSH}^A .
13. If Alice passes the $\text{TiltedCHSH}(\Gamma_{\text{CHSH}}^A, \text{Alice})$ test then both Alice and Bob proceed to the next phase where Alice self-tests her measurement device.

Alice's Measurement Device Verification Phase:

- Alice and Bob consider the rest $(\mathcal{K} - \gamma_1 \mathcal{K})$ samples and construct a set Γ_{test}
- For $1 \leq i \leq |\Gamma_{\text{test}}|$, Bob does the following.
 - Bob applies measurement operator B_0^0 or B_1^0 randomly on his particle of the shared state indexed by i and generates the output bit $b_i = 0$ and $b_i = 1$ respectively.
 - If the outcome of Bob's device for the shared state indexed by i is $b_i = 0$, Bob considers the raw key bit indexed by i as $R_i = 0$.
 - If the outcome of Bob's device for the shared state indexed by i is $b_i = 1$, Bob considers the raw key bit indexed by i as $R_i = 1$.
- Alice chooses $\gamma_2 |\Gamma_{\text{test}}|$ instances randomly from these $|\Gamma_{\text{test}}|$ states, constructs a set Γ_{POVM} with those samples and declares those instances (Note that no commitment scheme is required here like our previous proposal as in this modified scheme, Alice is performing optimal individual measurements at her end. So, Alice can't retrieve any additional bits in the *shared key generation phase* by performing any other measurements. Alice can at most perform joint measurements to retrieve the final key bits instead of the individual raw key bits. However, these optimal joint measurements are already shown to be inconclusive [63, 15] and are of no use to Alice).
- Alice first performs $\text{KeyGenAlice}(\Gamma_{\text{POVM}})$, according to the procedure introduced in algorithm 8 for the set Γ_{POVM} .
- Bob and Alice then perform $\text{POVMtestAlice}(\Gamma_{\text{POVM}})$, according to the procedure introduced in algorithm 9 for the same set Γ_{POVM} .

- If Alice and Bob pass the $\text{POVMtestAlice}(\Gamma_{\text{POVM}})$ then they proceed to the next phase of the protocol where they generate the shared key.

Key Generation Phase:

- Alice and Bob consider the rest ($|\Gamma_{\text{test}}| - |\Gamma_{\text{POVM}}|$) samples, construct a set Γ_{QPQ} with those instances and do the following.
 1. Alice performs $\text{KeyGenAlice}(\Gamma_{\text{QPQ}})$, as mentioned in algorithm 8 for the set Γ_{QPQ} .
 2. Bob already generates the raw key bits for each of the instances in Γ_{QPQ} .

Private Query Phase:

- Alice and Bob then use classical methods to process the raw key and move to the private query phase described in [117] (detailed procedure is already mentioned in the previous proposal of this chapter).

A visual illustration of different steps of this full device-independent proposal for a modification of the QPQ scheme [117] is depicted in Figure 5-5.

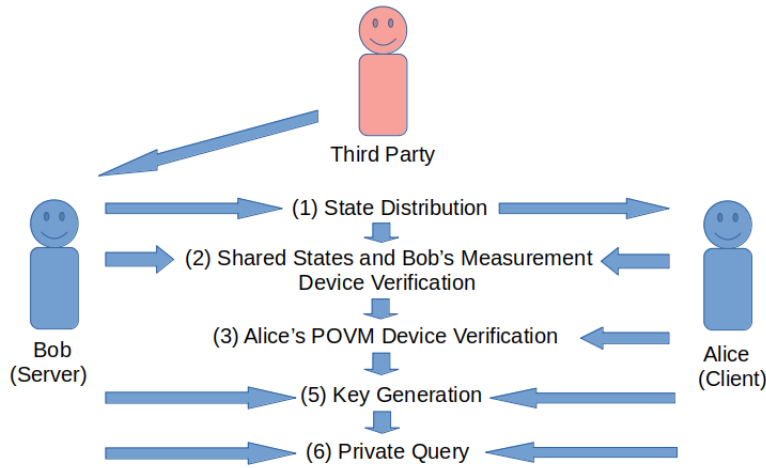


Figure 5-5: Visual representation of our modified DI-QPQ scheme

5.5.2 Analysis of the modified scheme

Here, we address the functionality of this proposal. At first, we prove its correctness, and next, we discuss the security aspects.

Correctness of our modified scheme

First, we prove the correctness of this modified scheme.

Theorem 11. *If the modified proposal is implemented honestly, then after the key generation phase, Alice is able to retrieve only $(1 - \cos \theta)$ fraction of the entire raw key.*

Proof. After the *shared key generation phase*, Bob and Alice share $|\Gamma_{\text{QPQ}}|$ raw key bits. These raw key bits were generated from $|\Gamma_{\text{QPQ}}|$ copies of shared entangled states which are of the form

$$\frac{1}{\sqrt{2}}(|0\rangle|\phi_0\rangle + |1\rangle|\phi_1\rangle)$$

where, $|\phi_0\rangle = \cos \frac{\theta}{2}|0\rangle + \sin \frac{\theta}{2}|1\rangle$ and $|\phi_1\rangle = \cos \frac{\theta}{2}|0\rangle - \sin \frac{\theta}{2}|1\rangle$. Here θ may vary from 0 to $\frac{\pi}{2}$.

Bob and Alice generate these $|\Gamma_{\text{QPQ}}|$ many raw key bits as follows-

For each of the states in the set Γ_{QPQ} , Bob measures his qubits in $\{|0\rangle, |1\rangle\}$ basis. For any i -th instance, if Bob receives the outcome $|0\rangle$, he considers $R_i = 0$ and $R_i = 1$ otherwise (i.e., for outcome $|1\rangle$).

Now, Alice understands that after Bob's measurement, her qubits corresponding to each of the shared states collapse to either $|\phi_0\rangle$ or $|\phi_1\rangle$. However, to obtain the value of the raw key bit, Alice has to distinguish these two states conclusively. As, $|\phi_0\rangle$ and $|\phi_1\rangle$ are non-orthogonal states (when $\theta \neq \frac{\pi}{2}$), Alice cannot distinguish these two states with certainty.

According to the strategy mentioned in this modified protocol, Alice chooses the POVM $\{D_0, D_1, D_2\}$ for measurement. After measurement, if Alice receives the outcome D_0 for i -th instance, she concludes that Bob's corresponding measurement outcome was $|0\rangle$. In such case, Alice concludes that $R_{\mathcal{A}_i} = 0$. Similarly, if Alice receives the outcome D_1 for i -th instance, she concludes that Bob's corresponding measurement outcome was $|1\rangle$. In such a case, Alice concludes that $R_{\mathcal{A}_i} = 1$. However, if the measurement outcome is D_2 , then Alice remains inconclusive about the value of the raw key bit.

Now, we calculate the success probability of Alice in guessing each R_i correctly. Let us assume that $\Pr(D_j|\phi_i)$ denotes the corresponding success probability of getting the result D_j when the given state is $|\phi_i\rangle$ i.e.,

$$\Pr(D_j|\phi_i) = \langle \phi_i | D_j | \phi_i \rangle.$$

We now calculate the corresponding success probabilities of getting different results for the states $|\phi_0\rangle$ and $|\phi_1\rangle$.

For $|\phi_0\rangle$, the success probabilities will be

$$\begin{aligned}
\Pr(D_0||\phi_0\rangle) &= \langle\phi_0| D_0|\phi_0\rangle \\
&= (1 - \cos\theta) \\
\Pr(D_1||\phi_0\rangle) &= \langle\phi_0| D_1|\phi_0\rangle \\
&= 0 \\
\Pr(D_2||\phi_0\rangle) &= \langle\phi_0| D_2|\phi_0\rangle \\
&= \cos\theta
\end{aligned}$$

Similarly, for the state $|\phi_1\rangle$, the success probabilities will be

$$\begin{aligned}
\Pr(D_0||\phi_1\rangle) &= \langle\phi_1| D_0|\phi_1\rangle \\
&= 0 \\
\Pr(D_1||\phi_1\rangle) &= \langle\phi_1| D_1|\phi_1\rangle \\
&= (1 - \cos\theta) \\
\Pr(D_2||\phi_1\rangle) &= \langle\phi_1| D_2|\phi_1\rangle \\
&= \cos\theta
\end{aligned}$$

We formalize all the conditional probabilities in Table 5.1. Thus, the success probability of Alice in guessing R_i of Bob can be written as

$$\begin{aligned}
&\Pr(R_{A_i} = R_i) \\
&= \Pr(R_{A_i} = 0, R_i = 0) + \Pr(R_{A_i} = 1, R_i = 1) \\
&= (1 - \cos\theta).
\end{aligned}$$

So, the success rate of Alice in guessing each bit of Bob's raw key in this modified proposal's *shared key generation phase* is $(1 - \cos\theta)$, meaning she can determine on average $(1 - \cos\theta)$ fraction of bits from the entire raw key with certainty (about the positions of the correctly predicted bits). □

Estimation of parameters for private query phase

Considering the honest implementation of this modified scheme, here we determine the values for different parameters to ensure both the privacy of the user and the privacy of the database owner.

Estimation of θ for security purpose :

Like our previous full DI version of [117], here also the server Bob wants the client Alice to know not more than one final key bit. In this modified proposal, the server

Bob has a raw key with kN many bits and the client Alice can correctly guess each of those bits with likelihood around $(1 - \cos \theta)$. So, the expected number of raw key bits that Alice can know in $(1 - \cos \theta)kN$.

Then each of Alice and Bob XOR k raw key bits to construct every final key bit at their end. So, Alice can correctly guess every bit of Bob's final key with probability around $(1 - \cos \theta)^k$.

Now, if $F_{\mathcal{A}}$ denotes Alice's known final key bits then we can conclude that the expected value of $F_{\mathcal{A}}$ will be,

$$E[F_{\mathcal{A}}] \approx (1 - \cos \theta)^k N. \quad (5.22)$$

In this modified DI scheme, for dishonest Alice to pass DI testing phases, she must measure correctly for all instances. Moreover, it is known that the optimal probability in distinguishing two non orthogonal states is $(1 - \cos \theta)$, which means dishonest Alice's probability of correctly guessing a raw key bit and a final key bit without causing the scheme to terminate is capped at $(1 - \cos \theta)$ and $(1 - \cos \theta)^k$, respectively. That means, when the protocol doesn't terminate, the expected number of correctly guessed final key bits by dishonest Alice is at most limited by,

$$E[F_{\mathcal{A}^*}] \leq (1 - \cos \theta)^k N. \quad (5.23)$$

Like the Yang et al. [117] QPQ scheme, here also the database is encrypted with the final key by performing bitwise XOR. Hence, in non abort scenario, the expected maximum number of correctly guessed data bits by dishonest Alice in a single query is limited to $(1 - \cos \theta)^k N$. i.e.,

$$E[D_{\mathcal{A}^*}] \leq (1 - \cos \theta)^k N. \quad (5.24)$$

Now, like the previous proposal, here also for the protocol to continue, Alice must know atleast one final key bit, while Bob wants Alice to know less than two final key bits i.e.,

$$1 \leq E[F_{\mathcal{A}}] < 2.$$

This implies that,

$$\begin{aligned} 1 &\leq (1 - \cos \theta)^k N < 2 \\ \frac{1}{N} &\leq (1 - \cos \theta)^k < \frac{2}{N}. \end{aligned} \quad (5.25)$$

These results boil down to the following conclusion.

Corollary 11. *To ensure that the client Alice only knows less than two final key bits*

and the scheme doesn't terminate in this modified proposal, the server Bob must select the values of θ and the parameter k such that,

$$\frac{1}{N} \leq (1 - \cos \theta)^k < \frac{2}{N}.$$

Estimation of P_a and P_c for security purpose:

Proceeding to the similar way as discussed in corollary 6, here we can assert that Alice can't guess any final key bit with probability

$$\begin{aligned} \Pr(\text{the protocol aborts}) &\approx [1 - (1 - \cos \theta)^k]^N \\ &\approx e^{-(1 - \cos \theta)^k N}. \end{aligned} \quad (5.26)$$

So, for the parameter P_a , we get the following upper bound for this modified scheme.

$$P_a \leq e^{-(1 - \cos \theta)^k N}. \quad (5.27)$$

If Bob sets θ so that $(1 - \cos \theta)^k = \frac{1}{N}$, then equation 5.25 and 5.27 yield

$$\boxed{P_a \leq e^{-1}}. \quad (5.28)$$

This implies that this modified proposal has a small P_a value. So, the probability of the protocol not aborting in the honest scenario is,

$$\begin{aligned} \Pr(\text{protocol doesn't terminate in honest scenario}) \\ \geq (1 - e^{-1}). \end{aligned} \quad (5.29)$$

Hence, this modified proposal has a high probability of not aborting in the honest scenario.

Like the previous scheme, here also (proceeding to the similar way) one can achieve the below mentioned bound on P_c for this modified scheme.

$$\boxed{P_c \geq [1 - \exp(-2\epsilon^2 N)]}, \quad (5.30)$$

where $\epsilon \leq \frac{1}{2}$ for security purpose.

We now proceed to the security aspects of this modified proposal.

Security in device independent scenario

In this subsection, we discuss about the DI security of this modified QPQ proposal. Based on the results obtained from Theorem 12 and Theorem 13, here we conclude

about the DI security of this modified QPQ scheme.

Theorem 12 (DI testing of shared states and Bob’s measurement devices). *In the TiltedCHSH test of the source device and Bob’s measurement device verification phase of our modified proposal, either the devices achieve $\beta_{\mathcal{B}} = \frac{4}{\sqrt{1+\sin^2\theta}}$ for both Alice and Bob (i.e., the states provided by the third party are identical with the actual states and Bob’s measurement device measures correctly in the $\{|0\rangle, |1\rangle\}$ basis) or the scheme is likely to abort with high probability (as the limit approaches infinity).*

Proof. This proof is same as the proof of theorem 8. □

So, Theorem 12 guarantees that either the states shared between Alice and Bob is of the specified form and Bob’s measurement device measures correctly in $\{|0\rangle, |1\rangle\}$ basis or this modified scheme aborts with high likelihood in the long run. The next testing for full DI certification is done in *Alice’s measurement device verification phase*. This phase basically guarantees the functionality of Alice’s POVM device. They lead to this phase whenever both of them successfully pass the first DI testing phase. In this phase, Alice performs the POVM measurement $D = \{D_0, D_1, D_2\}$ on the chosen states. From the measurement outcome, Alice computes the value of the parameter Ω and checks whether this value is equal to $\frac{2\sin^2\theta}{(1+\cos\theta)}$. Theorem 13 guarantees that either Alice measures correctly using the measurement device $\{D_0, D_1, D_2\}$ (i.e., the devices achieve $\Omega = \frac{2\sin^2\theta}{(1+\cos\theta)}$) or this modified proposal terminates with high probability (as the limit approaches infinity).

Theorem 13 (DI Testing of Alice’s POVM D). *POVMtestAlice either results in a high probability of termination of this modified proposal (as the limit approaches infinity), or it guarantees that Alice’s measurement devices attain $\Omega = \frac{2\sin^2\theta}{(1+\cos\theta)}$, meaning they are of this specified form (up to a local unitary),*

$$\begin{aligned} D_0 &= \frac{1}{(1+\cos\theta)}(|\phi_1^\perp\rangle\langle\phi_1^\perp|) \\ D_1 &= \frac{1}{(1+\cos\theta)}(|\phi_0^\perp\rangle\langle\phi_0^\perp|) \\ D_2 &= \mathbb{I} - D_0 - D_1, \end{aligned}$$

where $|\phi_1^\perp\rangle = (\sin\frac{\theta}{2}|0\rangle + \cos\frac{\theta}{2}|1\rangle)$ and $|\phi_0^\perp\rangle = (\sin\frac{\theta}{2}|0\rangle - \cos\frac{\theta}{2}|1\rangle)$.

The detailed proof of this theorem is mentioned later in Section 5.8. In the proof, we consider a general form of a single qubit three outcome POVM $\{D_0, D_1, D_2\}$ and show that if the input states are chosen randomly between $|\phi_0\rangle = (\cos\frac{\theta}{2}|0\rangle + \sin\frac{\theta}{2}|1\rangle)$ and $|\phi_1\rangle = (\cos\frac{\theta}{2}|0\rangle - \sin\frac{\theta}{2}|1\rangle)$ then either $\Omega = \frac{2\sin^2\theta}{(1+\cos\theta)}$ i.e., $\{D_0, D_1, D_2\}$ are of the specified form as mentioned in POVMtestAlice or this modified proposal terminates with high likelihood(as the limit approaches infinity).

Note that in our proof, we have not imposed any dimension bound like the self-testing of POVM in a prepare and measure scenario in [106]. So, the devices that

perform a Neumark dilation of this mentioned POVM (i.e., the equivalent larger projective measurement on both the original state and some ancilla system instead of the actual POVM measurement) could still achieve the intended value of Ω . But both of these operations produce the same output probabilities, which is sufficient for the purposes of this work.

Like the previous full DI proposal of the QPQ scheme [117], here also one can argue in a similar way that *this modified scheme either terminates with high probability (as the limit approaches infinity) or it certifies that the devices in this modified QPQ proposal achieve the desired values of the parameters β_B and Ω in the TiltedCHSH test and POVMtestAlice respectively.*

Security of database against dishonest Alice

Here, we estimate the amount of raw key bits guessed by dishonest Alice during the *shared key generation phase* of this modified scheme. Similar to the result in Theorem 10, here also we can conclude the following.

Theorem 14. *For this modified DI-QPQ scheme, in the absence of POVMtestAlice, dishonest Alice can retrieve, at most, $(\frac{1}{2} + \frac{1}{2} \sin \theta)$ fraction of the entire raw key, inconclusively (i.e., the indices of the correctly guessed bits are unknown), during the key generation phase.*

The proof is exactly the same as the proof of Theorem 5 in [15].

In this modified DI-QPQ proposal, Alice performs a particular POVM measurement to distinguish the non-orthogonal states at her end which is also the optimal measurement to distinguish that specified non-orthogonal states. Because of this specific measurement, we can get a bound on the number of raw key bits guessed (on average) by dishonest Alice in this proposed scheme.

Lemma 5. *Either our modified protocol terminates with high likelihood in the long run, or dishonest Alice (\mathcal{A}^*) can retrieve (on average) $(1 - \cos \theta)$ fraction from the entire raw key after the key generation phase of this modified scheme.*

The proof of this Lemma is based on the Theorem 11 which establishes the correctness of this modified scheme.

One can also note that this $(1 - \cos \theta)$ is the optimal probability (this optimality is proven in [62]) of success in distinguishing two non-orthogonal states with certainty (which is the main objective of the client Alice here in this modified proposal).

Here, for this modified scheme, equation 5.24 and definition 3 yield the following bound on τ .

Corollary 12. *In the case of dishonest Alice and honest Bob, either this modified proposal will likely abort (as the limit approaches infinity), or dishonest Alice will, on average, be able to obtain τ fraction of bits from the entire final key, where*

$$\tau \leq (1 - \cos \theta)^k. \tag{5.31}$$

By using the upper bound from equation 5.25 in place of $(1 - \cos \theta)^k$, we can obtain the following bound on τ .

$$\boxed{\tau < \frac{2}{N}}. \quad (5.32)$$

It shows that this modified proposal results in τ being significantly smaller than N .

Security of user against dishonest Bob

In this subsection, we estimate the number of indices that dishonest Bob can correctly guess from \mathcal{I}_l (the query index set of Alice) after successfully passing the *shared key generation phase* of this modified scheme. Similar to the result in Lemma 4, here also we can conclude the following.

Lemma 6. *Dishonest Bob can correctly predict a maximum of $\frac{l}{N}$ fraction of the indices from the query set \mathcal{I}_l for this modified proposal, i.e., for a particular index i ,*

$$\Pr(\text{Bob correctly guesses } i \in \mathcal{I}_l) \leq \frac{l}{N}.$$

The proof of Lemma 6 is identical to the proof of Lemma 4.

Like the discussion in corollary 10, bounds on δ and P_u can also be obtained for this modified proposal.

Corollary 13. *In dishonest Bob and honest Alice scenario of this modified DI-QPQ proposal, the scheme will either abort with high likelihood (as the limit approaches infinity), or dishonest Bob will be able to correctly predict, on average, δ fraction of indices from \mathcal{I}_l (the query index set of Alice) where,*

$$\boxed{\delta \leq \frac{1}{l^{(n-1)}}}, \quad (5.33)$$

where n is a positive integer such that $n > 1$. From this relation, one can conclude that δ is smaller than l for this modified proposal.

Now we mention the detailed proof of our results (i.e., theorems) in subsequent sections.

5.6 Statement and proof of Theorem 8

Statement of Theorem 8: In the TiltedCHSH test of the *source device and Bob's measurement device verification phase*, either the devices achieve $\beta_{\mathcal{B}} = \frac{4}{\sqrt{1+\sin^2 \theta}}$ for both Alice and Bob (i.e., the states provided by the third party are identical with the actual states as mentioned in the QPQ scheme [117] and Bob's measurement device measures correctly in the $\{|0\rangle, |1\rangle\}$ basis) or the scheme is likely to abort with high probability (as the limit approaches infinity).

Proof. Here we prove the result considering that the game is played at the party \mathcal{P} 's end (one can replace \mathcal{P} with Alice or Bob for the specific instances). Suppose, the first measurement operators of \mathcal{P} are $\{B_b^y\}_{y,b \in \{0,1\}}$, for the input y and the output b and the second measurement operators of \mathcal{P} are $\{A_a^x\}_{x,a \in \{0,1\}}$, for the input x and the output a . Here, \mathcal{P} 's observable corresponding to the input $y \in \{0,1\}$ is,

$$B_y = \sum_{b \in \{0,1\}} (-1)^b B_b^y. \quad (5.34)$$

Similarly, \mathcal{P} 's observable corresponding to the input $x \in \{0,1\}$ is,

$$A_x' = \sum_{a \in \{0,1\}} (-1)^a A_a^x. \quad (5.35)$$

Note that, in the TiltedCHSH test, the fraction $\beta_{\mathcal{B}}$ is being computed as follows,

$$\beta_{\mathcal{B}} = \alpha_{\mathcal{B}} \sum_{a \in \{0,1\}} (-1)^a \langle \psi_{\mathcal{B}\mathcal{A}} | \mathbb{I} \otimes A_a^{\prime 0} | \psi_{\mathcal{B}\mathcal{A}} \rangle \quad (5.36)$$

$$+ \sum_{x,y,a,b \in \{0,1\}} (-1)^{d_{xyab}} \langle \psi_{\mathcal{B}\mathcal{A}} | B_b^y \otimes A_a^x | \psi_{\mathcal{B}\mathcal{A}} \rangle \quad (5.37)$$

$$= [\langle \psi_{\mathcal{B}\mathcal{A}} | W_{\mathcal{B}}^1 | \psi_{\mathcal{B}\mathcal{A}} \rangle + \langle \psi_{\mathcal{B}\mathcal{A}} | W_{\mathcal{B}}^2 | \psi_{\mathcal{B}\mathcal{A}} \rangle] \quad (5.38)$$

$$= \langle \psi_{\mathcal{B}\mathcal{A}} | W_{\mathcal{B}} | \psi_{\mathcal{B}\mathcal{A}} \rangle, \quad (5.39)$$

where $W_{\mathcal{B}}^1 := \alpha_{\mathcal{B}} \sum_{a \in \{0,1\}} (-1)^a \mathbb{I} \otimes A_a^{\prime 0}$, $W_{\mathcal{B}}^2 := \left(\sum_{x,y,a,b \in \{0,1\}} (-1)^{d_{xyab}} B_b^y \otimes A_a^x \right)$ are the two operators corresponding to $\beta_{\mathcal{B}}$ of the TitedCHSH test and $W_{\mathcal{B}} := W_{\mathcal{B}}^1 + W_{\mathcal{B}}^2$. We can rewrite the expression of $W_{\mathcal{B}}^1$ in the following way.

$$\begin{aligned} W_{\mathcal{B}}^1 &= \alpha_{\mathcal{B}} \sum_{a \in \{0,1\}} (-1)^a \mathbb{I} \otimes A_a^{\prime 0} \\ &= \alpha_{\mathcal{B}} (\mathbb{I} \otimes A_0^{\prime 0} - \mathbb{I} \otimes A_1^{\prime 0}) \\ &= \alpha_{\mathcal{B}} [\mathbb{I} \otimes (A_0^{\prime 0} - A_1^{\prime 0})]. \end{aligned}$$

By substituting the value of $(A_0^{\prime 0} - A_1^{\prime 0})$ from the equation 5.35 on the right-hand side of the above expression we get,

$$W_{\mathcal{B}}^1 = \alpha_{\mathcal{B}} (\mathbb{I} \otimes A_0'). \quad (5.40)$$

Similarly, We can also rewrite the expression of $W_{\mathcal{B}}^2$ in following way.

$$\begin{aligned}
W_{\mathcal{B}}^2 &= \left(\sum_{\substack{x=0 \\ y,a,b \in \{0,1\}}} (-1)^{d_{xyab}} B_b^y \otimes A_a'^0 \right) + \\
&\quad \left(\sum_{\substack{x=1 \\ y,a,b \in \{0,1\}}} (-1)^{d_{xyab}} B_b^y \otimes A_a'^1 \right) \\
&= W_{\mathcal{B}}^{02} + W_{\mathcal{B}}^{12}, \tag{5.41}
\end{aligned}$$

where $W_{\mathcal{B}}^{02} := \left(\sum_{y,a,b \in \{0,1\}} \sum_{x=0} (-1)^{d_{xyab}} B_b^y \otimes A_a'^0 \right)$ and $W_{\mathcal{B}}^{12} := \left(\sum_{y,a,b \in \{0,1\}} \sum_{x=1} (-1)^{d_{xyab}} B_b^y \otimes A_a'^1 \right)$. Note that, we can simplify further the expression of $W_{\mathcal{B}}^{02}$ in the following way.

$$\begin{aligned}
W_{\mathcal{B}}^{02} &= \sum_{\substack{x=0 \\ y,a,b \in \{0,1\}}} (-1)^{d_{xyab}} B_b^y \otimes A_a'^0 \\
&= \sum_{\substack{x=0 \\ y,a,b \in \{0,1\} \\ a \oplus b = 0}} B_b^y \otimes A_a'^0 - \sum_{\substack{x=0 \\ y,a,b \in \{0,1\} \\ a \oplus b \neq 0}} B_b^y \otimes A_a'^0 \\
&= (B_0^0 \otimes A_0'^0 + B_0^1 \otimes A_0'^0 + B_1^0 \otimes A_1'^0 + B_1^1 \otimes A_1'^0) - \\
&\quad (B_1^0 \otimes A_0'^0 + B_1^1 \otimes A_0'^0 + B_0^0 \otimes A_1'^0 + B_0^1 \otimes A_1'^0) \\
&= [(B_0^0 - B_1^0) \otimes A_0'^0 - (B_0^0 - B_1^0) \otimes A_0'^0 + \\
&\quad (B_0^1 - B_1^1) \otimes A_0'^0 - (B_0^1 - B_1^1) \otimes A_1'^0] \\
&= [(B_0^0 - B_1^0) \otimes (A_0'^0 - A_1'^0) + \\
&\quad (B_0^1 - B_1^1) \otimes (A_0'^0 - A_1'^0)] \\
&= [(B_0^0 - B_1^0) + (B_0^1 - B_1^1)] \otimes (A_0'^0 - A_1'^0).
\end{aligned}$$

By substituting the values of $(A_0'^0 - A_1'^0)$, $(B_0^0 - B_1^0)$ and $(B_0^1 - B_1^1)$ from the equation 5.35 and the equation 5.34 on the right-hand side of the above expression we get,

$$W_{\mathcal{B}}^{02} = (B_0 + B_1) \otimes A_0'. \tag{5.42}$$

Using similar approach we get the following simplified version of the expression $W_{\mathcal{B}}^{12}$.

$$W_{\mathcal{B}}^{12} = (B_0 - B_1) \otimes A_1'. \tag{5.43}$$

By substituting the values of $W_{\mathcal{B}}^{02}$ and $W_{\mathcal{B}}^{12}$ from the equation 5.42 and the equa-

tion 5.43 to the equation 5.41 we get,

$$W_{\mathcal{B}}^2 = (B_0 + B_1) \otimes A'_0 + (B_0 - B_1) \otimes A'_1. \quad (5.44)$$

So, the right-hand side of the TiltedCHSH operator $W_{\mathcal{B}}$ is of the form,

$$W_{\mathcal{B}} = \alpha_{\mathcal{B}}(\mathbb{I} \otimes A'_0) + (B_0 + B_1) \otimes A'_0 + (B_0 - B_1) \otimes A'_1. \quad (5.45)$$

Note that this TiltedCHSH operator is exactly of the same form as the Tilted-CHSH operator mentioned in [8]. Also, the states mentioned in our protocol can be obtained from the non-maximally entangled states mentioned in [8] by just applying a local unitary (Hadamard gate) on the first qubit of the states mentioned in [8]. So, by following the same strategy as mentioned in [8], we can derive the following upper bound on the value of $\beta_{\mathcal{B}}$.

$$\beta_{\mathcal{B}} \leq \frac{4}{\sqrt{1 + \sin^2 \theta}}. \quad (5.46)$$

One can easily check that for the TiltedCHSH test, the observables of \mathcal{P} are of the following form.

$$B_0 = \sigma_z \quad B_1 = \sigma_x \quad (5.47)$$

$$A'_0 = \cos \mu \sigma_z + \sin \mu \sigma_x \quad A'_1 = \cos \mu \sigma_z - \sin \mu \sigma_x. \quad (5.48)$$

It is already mentioned in [14] that the maximum value of the TiltedCHSH operator (here $\beta_{\mathcal{B}} = \frac{4}{\sqrt{1 + \sin^2 \theta}}$) certifies that the states are of the form $\cos \frac{\theta}{2} |00\rangle + \sin \frac{\theta}{2} |11\rangle$ and the observables of \mathcal{P} 's are of the same form as mentioned in our TiltedCHSH test. As the states shared in our scheme is just a local isometry of the states mentioned in [14], we can easily conclude from the results mentioned in [14] that the maximum value of $\beta_{\mathcal{B}}$ (i.e., $\beta_{\mathcal{B}} = \frac{4}{\sqrt{1 + \sin^2 \theta}}$) certifies the states in our scheme along with the standard basis of Bob's measurement device. According to our DI proposal, whenever the devices don't achieve the value $\beta_{\mathcal{B}} = \frac{4}{\sqrt{1 + \sin^2 \theta}}$, the protocol aborts. This concludes the proof. □

5.7 Statement and proof of Theorem 9

Statement of Theorem 9: In OBStestAlice, either Alice's measurement devices achieve the value of the parameter $\beta_{\mathcal{A}} = \frac{1}{2 \sin \theta}$ (i.e., her devices correctly measure in $\{|\phi_0\rangle, |\phi_0^\perp\rangle\}$ and $\{|\phi_1\rangle, |\phi_1^\perp\rangle\}$ basis) or the protocol terminates with a high likelihood of failure (as the limit approaches infinity).

Proof. It is already mentioned in the proof of theorem 8 that Alice's measurement operators are $\{A_a^x\}_{x,a \in \{0,1\}}$, corresponding to the input x and output a and Bob's

measurement operators are $\{B_b^y\}_{y,b \in \{0,1\}}$, corresponding to the input y and output b . So, Alice's observable, corresponding to the input $x \in \{0,1\}$ is,

$$A_x = \sum_{a \in \{0,1\}} (-1)^a A_a^x. \quad (5.49)$$

Similarly, Bob's observable corresponding to the input $y \in \{0,1\}$ is,

$$B_y = \sum_{b \in \{0,1\}} (-1)^b B_b^y. \quad (5.50)$$

Note that in the OBStestAlice, the fraction $\beta_{\mathcal{A}}$ is being computed as follows,

$$\beta_{\mathcal{A}} = \frac{1}{4} \sum_{x,y,a,b \in \{0,1\}} (-1)^{d'_{xyab}} \alpha_{\mathcal{A}}^{1 \oplus y} \langle \psi | B_b^y \otimes A_a^x | \psi \rangle \quad (5.51)$$

$$= \frac{1}{4} \langle \psi | W_{\mathcal{A}} | \psi \rangle, \quad (5.52)$$

where $W_{\mathcal{A}} := \left(\sum_{x,y,a,b \in \{0,1\}} (-1)^{d'_{xyab}} \alpha_{\mathcal{A}}^{1 \oplus y} B_b^y \otimes A_a^x \right)$ which is the operator corresponding to $\beta_{\mathcal{A}}$ of OBStestAlice. Now, proceeding like the similar way as mentioned in the derivation of the simplified form for operator $W_{\mathcal{B}}^2$ in the proof of theorem 8, here we can get the following expression of $W_{\mathcal{A}}$.

$$W_{\mathcal{A}} = \alpha_{\mathcal{A}} B_0 \otimes (A_0 + A_1) + B_1 \otimes (A_0 - A_1). \quad (5.53)$$

Note that, the right-hand side of the OBStestAlice operator $W_{\mathcal{A}}$ is almost of the same form as the tiltedCHSH operator, described in [65].

So the expression of $W_{\mathcal{A}}^2$ can be written as,

$$\begin{aligned} W_{\mathcal{A}}^2 &= \alpha_{\mathcal{A}}^2 B_0^2 \otimes (A_0^2 + A_1^2 + \{A_0, A_1\}) \\ &\quad + B_1^2 \otimes (A_0^2 + A_1^2 - \{A_0, A_1\}) \\ &= (\alpha_{\mathcal{A}}^2 B_0^2 + B_1^2 + \alpha_{\mathcal{A}} \{B_0, B_1\}) \otimes A_0^2 \\ &\quad + (\alpha_{\mathcal{A}}^2 B_0^2 + B_1^2 - \alpha_{\mathcal{A}} \{B_0, B_1\}) \otimes A_1^2 \\ &\quad + (\alpha_{\mathcal{A}}^2 B_0^2 - B_1^2) \otimes \{A_0, A_1\} - \alpha_{\mathcal{A}} [B_0, B_1] \otimes [A_0, A_1]. \end{aligned}$$

Using the property $B_j^2 \leq \mathbb{I}$, we can rewrite this expression as,

$$\begin{aligned} W_{\mathcal{A}}^2 &\leq [(\alpha_{\mathcal{A}}^2 + 1) \cdot \mathbb{I} + \alpha_{\mathcal{A}} \{B_0, B_1\}] \otimes A_0^2 \\ &\quad + [(\alpha_{\mathcal{A}}^2 + 1) \cdot \mathbb{I} - \alpha_{\mathcal{A}} \{B_0, B_1\}] \otimes A_1^2 \\ &\quad + \mathbb{I} \otimes (\alpha_{\mathcal{A}}^2 - 1) \{A_0, A_1\} - \alpha_{\mathcal{A}} [B_0, B_1] \otimes [A_0, A_1]. \end{aligned}$$

Since $-2 \cdot \mathbb{I} \leq \{B_0, B_1\} \leq 2 \cdot \mathbb{I}$, we have,

$$[(\alpha_{\mathcal{A}}^2 + 1).\mathbb{I} \pm \alpha_{\mathcal{A}}\{B_0, B_1\}] \geq 0.$$

We can use the property $A_k^2 \leq \mathbb{I}$ and get the following simplified expression.

$$W_{\mathcal{A}}^2 \leq 2(\alpha_{\mathcal{A}}^2 + 1).\mathbb{I} \otimes \mathbb{I} + \mathbb{I} \otimes (\alpha_{\mathcal{A}}^2 - 1)\{A_0, A_1\} \\ - \alpha[B_0, B_1] \otimes [A_0, A_1].$$

We can further upper bound the commutators by their matrix moduli and use the relation $|[B_0, B_1]| \leq 2.\mathbb{I}$ to get the following expression.

$$W_{\mathcal{A}}^2 \leq 2(\alpha_{\mathcal{A}}^2 + 1).\mathbb{I} \otimes \mathbb{I} + T_{\alpha_{\mathcal{A}}} \otimes \mathbb{I}, \quad (5.54)$$

where $T_{\alpha_{\mathcal{A}}} = (\alpha_{\mathcal{A}}^2 - 1)\{A_0, A_1\} + 2\alpha_{\mathcal{A}}|[A_0, A_1]|$

Now the expression of $T_{\alpha_{\mathcal{A}}}$ can also be upper bounded by upper bounding the anticommutators by its matrix modulus. So, the value of $T_{\alpha_{\mathcal{A}}}$ will be upper bounded by,

$$T_{\alpha_{\mathcal{A}}} \leq (\alpha_{\mathcal{A}}^2 - 1)|\{A_0, A_1\}| + 2\alpha_{\mathcal{A}}|[A_0, A_1]|.$$

Again one can easily check that,

$$|\{A_0, A_1\}|^2 + |[A_0, A_1]|^2 \\ = |A_0A_1 + A_1A_0|^2 + |A_0A_1 - A_1A_0|^2 \\ = (A_0A_1 + A_1A_0)^\dagger(A_0A_1 + A_1A_0) \\ + (A_0A_1 - A_1A_0)^\dagger(A_0A_1 - A_1A_0) \\ = 2(A_0A_1)^\dagger(A_0A_1) + 2(A_1A_0)^\dagger(A_1A_0). \quad (5.55)$$

Let us consider that the measurement operators are projective i.e., $(A_c^s)^2 = A_c^s$ and $(B_b^r)^2 = B_b^r$. Now for the projectors A_0^0 and A_1^0 , $(A_0^0 + A_1^0) = \mathbb{I}$. From this relation we can write,

$$(A_0^0 + A_1^0)(A_0^0 + A_1^0)^\dagger = \mathbb{I} \\ A_0^0.A_0^{0\dagger} + A_0^0.A_1^{0\dagger} + A_1^0.A_0^{0\dagger} + A_1^0.A_1^{0\dagger} = \mathbb{I} \\ (A_0^0 + A_1^0) + (A_0^0.A_1^{0\dagger} + A_1^0.A_0^{0\dagger}) = \mathbb{I}.$$

This implies,

$$(A_0^0.A_1^{0\dagger} + A_1^0.A_0^{0\dagger}) = 0.$$

Now $A_0 = (A_0^0 - A_1^0)$. From this we can get,

$$\begin{aligned} A_0 A_0^\dagger &= (A_0^0 - A_1^0)(A_0^0 - A_1^0)^\dagger \\ &= A_0^0.A_0^{0\dagger} - A_0^0.A_1^{0\dagger} - A_1^0.A_0^{0\dagger} + A_1^0.A_1^{0\dagger} \\ &= (A_0^0 + A_1^0) - (A_0^0.A_1^{0\dagger} + A_1^0.A_0^{0\dagger}) \\ &= \mathbb{I} + 0 = \mathbb{I}. \end{aligned}$$

Similarly, it can be shown that, $A_1 A_1^\dagger = A_1^\dagger A_1 = \mathbb{I}$.

So, from equation 5.55, we can write that for unitary observables A_0 and A_1 ,

$$\begin{aligned} |\{A_0, A_1\}|^2 + |[A_0, A_1]|^2 &= 2(A_0 A_1)^\dagger (A_0 A_1) \\ &\quad + 2(A_1 A_0)^\dagger (A_1 A_0) \\ &= 2\mathbb{I} + 2\mathbb{I} = 4\mathbb{I}. \end{aligned}$$

This implies,

$$|\{A_0, A_1\}| = \sqrt{4\mathbb{I} - |[A_0, A_1]|^2}.$$

So, the simplified expression of $T_{\alpha_{\mathcal{A}}}$ will be of the form

$$T_{\alpha_{\mathcal{A}}} = (\alpha_{\mathcal{A}}^2 - 1)\sqrt{4\mathbb{I} - |[A_0, A_1]|^2} + 2\alpha_{\mathcal{A}}|[A_0, A_1]|.$$

Now one can easily check that the value of $|[A_0, A_1]|$ for which the value of $T_{\alpha_{\mathcal{A}}}$ becomes maximum is $|[A_0, A_1]| = \frac{4\alpha_{\mathcal{A}}}{(\alpha_{\mathcal{A}}^2 + 1)}.\mathbb{I}$ and the corresponding value of $T_{\alpha_{\mathcal{A}}}$ is $2(\alpha_{\mathcal{A}}^2 + 1).\mathbb{I}$. This implies that,

$$T_{\alpha_{\mathcal{A}}} = 2(\alpha_{\mathcal{A}}^2 + 1).\mathbb{I}.$$

From this value of $T_{\alpha_{\mathcal{A}}}$ and from the expression of $W_{\mathcal{A}}^2$ mentioned in equation 5.54, we can easily write that the value of $W_{\mathcal{A}}$ is upper bounded by the following quantity.

$$W_{\mathcal{A}} \leq \sqrt{2(\alpha_{\mathcal{A}}^2 + 1)\mathbb{I} \otimes \mathbb{I} + T_{\alpha_{\mathcal{A}}} \otimes \mathbb{I}}, \quad (5.56)$$

where $T_{\alpha_{\mathcal{A}}} = 2(\alpha_{\mathcal{A}}^2 + 1).\mathbb{I}$.

Now, the value $\beta_{\mathcal{A}}$ obtained in OBStestAlice of our algorithm can be written alternatively as $\beta_{\mathcal{A}} = \frac{\text{Tr}(W_{\mathcal{A}}\rho_{\mathcal{B}\mathcal{A}})}{4}$ where $\rho_{\mathcal{B}\mathcal{A}}$ is the density matrix representation of the

shared states $|\psi\rangle_{\mathcal{B}\mathcal{A}}$ i.e., $\rho_{\mathcal{B}\mathcal{A}} = |\psi\rangle_{\mathcal{B}\mathcal{A}}\langle\psi|$. From this expression of $\beta_{\mathcal{A}}$, one can easily derive that the value of $\beta_{\mathcal{A}}^2$ is upper bounded by the following quantity.

$$\beta_{\mathcal{A}}^2 \leq \frac{\text{Tr}(W_{\mathcal{A}}^2 \rho_{\mathcal{B}\mathcal{A}})}{16}. \quad (5.57)$$

Now if we assume $t_{\alpha_{\mathcal{A}}} := \frac{1}{4}\text{Tr}(T_{\alpha_{\mathcal{A}}}\rho_{\mathcal{A}}) - \frac{1}{2}(\alpha_{\mathcal{A}}^2 - 1)$ (where $\rho_{\mathcal{A}}$ is the reduced state at Alice's side) then using this value of t_{α} along with the value of $W_{\mathcal{A}}$ obtained from expression 5.56 and the upper bound on the value of $\beta_{\mathcal{A}}^2$, we can write that the $\beta_{\mathcal{A}}$ value mentioned in OBStestAlice is upper bounded by the following quantity.

$$\beta_{\mathcal{A}} \leq \frac{\sqrt{\alpha_{\mathcal{A}}^2 + t_{\alpha_{\mathcal{A}}}}}{2}, \quad (5.58)$$

where, $t_{\alpha_{\mathcal{A}}} := \frac{1}{4}\text{Tr}(T_{\alpha_{\mathcal{A}}}\rho_{\mathcal{A}}) - \frac{1}{2}(\alpha_{\mathcal{A}}^2 - 1)$.

Now here, the observables are projective (i.e., $A_j^2 = \mathbb{I}$) and the anticommutator $\{A_0, A_1\}$ is a positive semi definite operator. Since we have already shown that the value of the anti-hermitian operator $[[A_0, A_1]]$ is $[[A_0, A_1]] = \frac{4\alpha_{\mathcal{A}}}{(\alpha_{\mathcal{A}}^2 + 1)}\mathbb{I}$ for the maximum value of $T_{\alpha_{\mathcal{A}}}$, the spectral decomposition of $[A_0, A_1]$ can be written as,

$$[A_0, A_1] = \frac{4\alpha_{\mathcal{A}} \cdot i}{(\alpha_{\mathcal{A}}^2 + 1)}(P_+^{\mathcal{A}} - P_-^{\mathcal{A}}),$$

for some orthogonal projectors $P_+^{\mathcal{A}}$ and $P_-^{\mathcal{A}}$ such that $(P_+^{\mathcal{A}} + P_-^{\mathcal{A}}) = \mathbb{I}$. As it is well-known that for projective observables, the commutator holds the property $A_0[A_0, A_1]A_0 = -[A_0, A_1]$, we can easily conclude that $A_0P_{\pm}^{\mathcal{A}}A_0 = P_{\mp}^{\mathcal{A}}$. Let us consider that $\{|e_j^0\rangle\}_j$ is an orthonormal basis for the support of $P_+^{\mathcal{A}}$ and $\{|e_j^1\rangle\}_j$ is an orthonormal basis for the support of $P_-^{\mathcal{A}}$ where $|e_j^1\rangle = A_0|e_j^0\rangle$. We define the unitary operator U_0 as

$$U_0|e_j^d\rangle = \frac{1}{\sqrt{2}}[|0\rangle + (-1)^d i|1\rangle]|j\rangle,$$

for $d \in \{0, 1\}$. Then we can easily verify that,

$$U_0[A_0, A_1]U_0^\dagger = \frac{4\alpha_{\mathcal{A}} \cdot i}{(\alpha_{\mathcal{A}}^2 + 1)}\sigma_Y \otimes \mathbb{I}.$$

Since $\{\mathbb{I}, \sigma_X, \sigma_Y, \sigma_Z\}$ constitute an operator basis for linear operators acting on \mathbb{C}^2 , without loss of generality we can write

$$U_0A_0U_0^\dagger = \mathbb{I} \otimes K_0 + \sigma_X \otimes K_x + \sigma_Y \otimes K_y + \sigma_Z \otimes K_z,$$

for some hermitian operator K_0, K_x, K_y, K_z . For projective observable A_0 , one can easily check that $\{A_0, [A_0, A_1]\} = 0$. This relation satisfies only when $K_0 = K_y = 0$. As $A_0^2 = \mathbb{I}$, K_x and K_z must satisfy the relation

$$K_x^2 + K_z^2 = \mathbb{I} \quad \text{and} \quad [K_x, K_z] = 0.$$

So, we can easily write K_x and K_z in the following form.

$$K_x = \sum_j \sin 2\gamma_j |j\rangle \langle j|$$

$$K_z = \sum_j \cos 2\gamma_j |j\rangle \langle j|,$$

for some angle γ_j and some orthonormal basis $\{|j\rangle\}$. This implies that,

$$U_0 A_0 U_0^\dagger = \sum_j (\sin \gamma_j \sigma_X + \cos \gamma_j \sigma_Z) \otimes |j\rangle \langle j|.$$

We now consider the following controlled unitary to align the qubit observables.

$$U_1 = \sum_j \exp(-i0 \cdot \sigma_Y) \otimes |j\rangle \langle j|.$$

Now for this defined unitary operator, one can easily check that,

$$U_1 U_0 A_0 U_0^\dagger U_1^\dagger = (\sin \gamma_j \sigma_X + \cos \gamma_j \sigma_Z) \otimes \mathbb{I}$$

$$U_1 U_0 [A_0, A_1] U_0^\dagger U_1^\dagger = \frac{4\alpha_{\mathcal{A}} \cdot i}{(\alpha_{\mathcal{A}}^2 + 1)} \sigma_Y \otimes \mathbb{I}.$$

Like observable A_0 , an analogous reasoning can also be applied for observable A_1 and from that, without loss of generality we can write

$$U_1 U_0 A_1 U_0^\dagger U_1^\dagger = \sigma_X \otimes K'_x + \sigma_Z \otimes K'_z.$$

Since the commutators are positive semi definite and the observables are projective, we can easily check that

$$\begin{aligned}\{A_0, A_1\} = |\{A_0, A_1\}| &= \sqrt{4\mathbb{I} - |[A_0, A_1]|^2} \\ &= \frac{2(\alpha_{\mathcal{A}}^2 - 1)}{(\alpha_{\mathcal{A}}^2 + 1)}\mathbb{I}.\end{aligned}$$

Now we define $2\gamma_j := \arccos\left(\frac{\alpha_{\mathcal{A}}^2 - 1}{\alpha_{\mathcal{A}}^2 + 1}\right) = 0$. From this relation, imposing consistency on the anticommutator, we get,

$$K'_x \sin \gamma_j + K'_z \cos \gamma_j = \cos 2\gamma_j. \quad (5.59)$$

On the other hand, imposing consistency on the commutator, we get,

$$K'_x \cos \gamma_j - K'_z \sin \gamma_j = -\sin 2\gamma_j. \quad (5.60)$$

Now, solving equation 5.59 and 5.60, we get,

$$K'_x = \sin \gamma_j \quad \text{and} \quad K'_z = \cos \gamma_j.$$

From the relation $2\gamma_j := \arccos\left(\frac{\alpha_{\mathcal{A}}^2 - 1}{\alpha_{\mathcal{A}}^2 + 1}\right) = 0$, we can get the value of $\alpha_{\mathcal{A}}$ which is

$$\alpha_{\mathcal{A}} = \cot \gamma_j.$$

For this value of $\alpha_{\mathcal{A}}$, we can easily derive that $t_{\alpha_{\mathcal{A}}} = 1$. This implies that the value of $\beta_{\mathcal{A}}$ corresponding to these observables A_0 and A_1 will be,

$$\beta_{\mathcal{A}} = \frac{1}{2 \sin \gamma_j}. \quad (5.61)$$

If we consider $U_{\mathcal{A}} = U_0^\dagger U_1^\dagger$ then the observables A_0 and A_1 will be of the form

$$\begin{aligned}A_0 &= U_{\mathcal{A}}(\cos \gamma_j \sigma_Z + \sin \gamma_j \sigma_X \otimes \mathbb{I})U_{\mathcal{A}}^\dagger \\ A_1 &= U_{\mathcal{A}}(\cos \gamma_j \sigma_Z - \sin \gamma_j \sigma_X \otimes \mathbb{I})U_{\mathcal{A}}^\dagger.\end{aligned}$$

Setting $\gamma_j = \theta$ shows that in OBStestAlice, if the value of the parameter $\beta_{\mathcal{A}}$ is equal to $\frac{1}{2 \sin \theta}$, then the measurement operators at Alice's side are same as the one described in the OBStestAlice. In our DI proposal, whenever the devices involved in OBStestAlice do not achieve the value $\beta_{\mathcal{A}} = \frac{1}{2 \sin \theta}$, the protocol aborts. This concludes the proof. \square

5.8 Statement and proof of Theorem 13

Statement of Theorem 13: POVMtestAlice either results in a high probability of termination of this modified proposal (as the limit approaches infinity), or it guarantees that Alice's measurement devices attain $\Omega = \frac{2\sin^2\theta}{(1+\cos\theta)}$, meaning they are of this specified form (up to a local unitary),

$$\begin{aligned} D_0 &= \frac{1}{(1+\cos\theta)}(|\phi_1^\perp\rangle\langle\phi_1^\perp|) \\ D_1 &= \frac{1}{(1+\cos\theta)}(|\phi_0^\perp\rangle\langle\phi_0^\perp|) \\ D_2 &= \mathbb{I} - D_0 - D_1, \end{aligned}$$

where $|\phi_1^\perp\rangle = (\sin\frac{\theta}{2}|0\rangle + \cos\frac{\theta}{2}|1\rangle)$ and $|\phi_0^\perp\rangle = (\sin\frac{\theta}{2}|0\rangle - \cos\frac{\theta}{2}|1\rangle)$.

Proof. In algorithm KeyGenAlice of this modified protocol, Alice applies the POVM D on a single qubit state ρ_{R_i} (where R_i is the raw key bit indexed by i at Bob's side). So, without any loss of generality we can assume that $D_i \in D$ has the following form.

$$D_i = \lambda_i(\mathbb{I} + \vec{d}_i \cdot \vec{\sigma}), \quad (5.62)$$

where $\vec{d}_i = [d_{i0}, d_{i1}, d_{i2}]$ and it is the Bloch vector with length at most one, $\vec{\sigma} = [\sigma_X, \sigma_Y, \sigma_Z]$ are the Pauli matrices and $\lambda_i \geq 0$.

In this case, one may wonder how we can fix the dimension of D_i here in the proof in DI scenario? The answer to this question is that here we are able to fix the dimension of D_i and choose this particular general form because of the tests mentioned earlier in the *source device and Bob's measurement device verification phase* (corresponding result mentioned in Theorem 12) which certifies that the states shared between Alice and Bob are of the specified form (upto a unitary) as mentioned in [117] and after Bob's projective measurements, the reduced states at Alice's side are one qubit states.

Now, the condition $\sum_{i=0}^2 D_i = \mathbb{I}$ leads us to the following relations.

$$\sum_{i=0}^2 \lambda_i = 1 \quad (5.63)$$

$$\sum_{i=0}^2 \lambda_i \vec{d}_i = 0. \quad (5.64)$$

In terms of Bloch vector we can rewrite ρ_0, ρ_1 in the following way.

$$\rho_0 = \frac{1}{2}(\mathbb{I} + \cos \theta \sigma_Z + \sin \theta \sigma_X) \quad (5.65)$$

$$\rho_1 = \frac{1}{2}(\mathbb{I} + \cos \theta \sigma_Z - \sin \theta \sigma_X). \quad (5.66)$$

In the algorithm `POVMtestAlice`, if Alice would like to maximize her winning probability then she needs to maximize the following expression.

$$\Omega = \sum_{R_i, R_{A_i} \in \{0,1\}} (-1)^{R_i \oplus R_{A_i}} \text{Tr}[D_{R_{A_i}} \rho_{R_i}]. \quad (5.67)$$

In terms of $\lambda_i, \vec{d}_i, \vec{\sigma}$ we have,

$$\begin{aligned} \text{Tr}[D_0 \rho_0] &= \lambda_0(1 + d_{00} \sin \theta + d_{02} \cos \theta) \\ \text{Tr}[D_0 \rho_1] &= \lambda_0(1 - d_{00} \sin \theta + d_{02} \cos \theta) \\ \text{Tr}[D_1 \rho_0] &= \lambda_1(1 + d_{10} \sin \theta + d_{12} \cos \theta) \\ \text{Tr}[D_1 \rho_1] &= \lambda_1(1 - d_{10} \sin \theta + d_{12} \cos \theta). \end{aligned}$$

In terms of $\lambda_i, \vec{d}_i, \vec{\sigma}$ can rewrite Ω as,

$$\begin{aligned} \Omega &= \lambda_0(1 + d_{00} \sin \theta + d_{02} \cos \theta) \\ &\quad + \lambda_1(1 - d_{10} \sin \theta + d_{12} \cos \theta) \\ &\quad - \lambda_0(1 - d_{00} \sin \theta + d_{02} \cos \theta) \\ &\quad - \lambda_1(1 + d_{10} \sin \theta + d_{12} \cos \theta). \end{aligned} \quad (5.68)$$

As both $\text{Tr}[D_0 \rho_1]$ and $\text{Tr}[D_1 \rho_0]$ are positive quantity, hence

$$\Omega \leq \lambda_0(1 + d_{00} \sin \theta + d_{02} \cos \theta) + \lambda_1(1 - d_{10} \sin \theta + d_{12} \cos \theta), \quad (5.69)$$

and this implies that for maximum value of Ω ,

$$\lambda_0(1 - d_{00} \sin \theta + d_{02} \cos \theta) = 0 \quad (5.70)$$

$$\lambda_1(1 + d_{10} \sin \theta + d_{12} \cos \theta) = 0. \quad (5.71)$$

As both of ρ_0, ρ_1 lie on the XZ plane and due to the freedom of global unitary, without loss of generality we can assume $d_{01} = d_{11} = d_{21} = 0$. Due to the positivity constraint ($D_i \geq 0$) we have,

$$d_{00}^2 + d_{02}^2 \leq 1 \quad (5.72)$$

$$d_{10}^2 + d_{12}^2 \leq 1 \quad (5.73)$$

$$d_{20}^2 + d_{22}^2 \leq 1. \quad (5.74)$$

Without any loss of generality we can assume that for the maximum value of Ω , $d_{00}^2 + d_{02}^2 = 1$. So, we can parameterize d_{00}, d_{02} as $\cos \alpha, \sin \alpha$ ($-2\pi \leq \alpha \leq 2\pi$). By substituting $d_{00} = \cos \alpha, d_{02} = \sin \alpha$ in equation 5.70 we get,

$$1 - \cos \alpha \sin \theta + \sin \alpha \cos \theta = 0.$$

This implies,

$$\sin(\theta - \alpha) = 1 = \sin \frac{\pi}{2}.$$

As $-2\pi \leq \alpha \leq 2\pi$, so $\sin(\theta - \alpha) = 1$ implies,

$$\begin{aligned} \theta - \alpha &= \frac{\pi}{2} \quad \text{and,} \\ \alpha &= \left(\theta - \frac{\pi}{2} \right). \end{aligned} \quad (5.75)$$

From the equation 5.75 we get,

$$\vec{d}_0 = [\sin \theta, 0, -\cos \theta]. \quad (5.76)$$

Similarly, for maximum value of Ω , $d_{10}^2 + d_{12}^2 = 1$. So, we can parameterize d_{10}, d_{12} as $\cos \alpha, \sin \alpha$ ($-2\pi \leq \alpha \leq 2\pi$). By substituting $d_{10} = \cos \alpha, d_{12} = \sin \alpha$ in equation 5.71 we get,

$$1 + \cos \alpha \sin \theta + \sin \alpha \cos \theta = 0.$$

This implies,

$$\sin(\theta + \alpha) = -1 = \sin \frac{3\pi}{2}.$$

As $-2\pi \leq \alpha \leq 2\pi$, so $\sin(\theta + \alpha) = -1$ implies,

$$\begin{aligned} \theta + \alpha &= \frac{3\pi}{2} \quad \text{and,} \\ \alpha &= \left(\frac{3\pi}{2} - \theta \right). \end{aligned} \quad (5.77)$$

From the equation 5.77 we get,

$$\vec{d}_1 = [-\sin \theta, 0, -\cos \theta]. \quad (5.78)$$

By substituting the expression of \vec{d}_0, \vec{d}_1 in equation 5.69 we get,

$$\Omega \leq (\lambda_0 + \lambda_1)(1 - \cos 2\theta). \quad (5.79)$$

Now again substituting the values of \vec{d}_0, \vec{d}_1 in equation 5.64 we get,

$$\lambda_0 \sin \theta - \lambda_1 \sin \theta + \lambda_2 d_{20} = 0 \quad (5.80)$$

$$-\lambda_0 \cos \theta - \lambda_1 \cos \theta + \lambda_2 d_{22} = 0. \quad (5.81)$$

Due to the constraint equation 5.74, similar to \vec{d}_0 and \vec{d}_1 , here also we parameterize the expression of d_{20}, d_{22} as $\sin \beta, \cos \beta$ respectively. By substituting $d_{20} = \sin \beta$ and $d_{22} = \cos \beta$ in the equations 5.80 and 5.81 we get,

$$\lambda_0 \sin \theta - \lambda_1 \sin \theta + \lambda_2 \sin \beta = 0 \quad (5.82)$$

$$-\lambda_0 \cos \theta - \lambda_1 \cos \theta + \lambda_2 \cos \beta = 0. \quad (5.83)$$

By solving equation 5.82 and equation 5.83 together with equation 5.63 we get,

$$\lambda_0 = \frac{\sin(\theta - \beta)}{[\sin(\theta + \beta) + \sin(\theta - \beta) + \sin 2\theta]} \quad (5.84)$$

$$\lambda_1 = \frac{\sin(\theta + \beta)}{[\sin(\theta + \beta) + \sin(\theta - \beta) + \sin 2\theta]}. \quad (5.85)$$

Hence,

$$\lambda_0 + \lambda_1 = \frac{\sin(\theta + \beta) + \sin(\theta - \beta)}{[\sin(\theta + \beta) + \sin(\theta - \beta) + \sin 2\theta]} \quad (5.86)$$

$$= \frac{\cos \beta}{(\cos \beta + \cos \theta)}. \quad (5.87)$$

According to equation 5.79, for getting a tight upper bound on Ω we need to maximize $(\lambda_0 + \lambda_1)$. By equating $\frac{d(\lambda_0 + \lambda_1)}{d\beta} = 0$ in equation 5.87 we get,

$$\frac{-\sin \beta \cos \theta}{(\cos \beta + \cos \theta)^2} = 0. \quad (5.88)$$

This implies,

$$\beta = 0. \quad (5.89)$$

It is also easy to check that for $\beta = 0$, the expression $\frac{d^2(\lambda_0 + \lambda_1)}{d\beta^2} < 0$. Hence, the expression $\lambda_0 + \lambda_1$ maximizes at the point $\beta = 0$. Substituting this relation in equations 5.84 and 5.85 we get,

$$\lambda_0 = \lambda_1 = \frac{1}{2(1 + \cos \theta)}. \quad (5.90)$$

By substituting the values of $\lambda_0 + \lambda_1$ in equation 5.63 we get,

$$\lambda_2 = \frac{\cos \theta}{1 + \cos \theta}. \quad (5.91)$$

Hence, we get,

$$\Omega \leq \frac{2 \sin^2 \theta}{(1 + \cos \theta)}, \quad (5.92)$$

and

$$D_0 = \frac{1}{2(1 + \cos \theta)} (\mathbb{I} + \sin \theta \sigma_X - \cos \theta \sigma_Z) \quad (5.93)$$

$$D_1 = \frac{1}{2(1 + \cos \theta)} (\mathbb{I} - \sin \theta \sigma_X - \cos \theta \sigma_Z) \quad (5.94)$$

$$D_2 = \frac{\cos \theta}{1 + \cos \theta} (\mathbb{I} + \sigma_Z). \quad (5.95)$$

We can rewrite the above expressions as follows,

$$D_0 = \frac{1}{(1 + \cos \theta)} (|\phi_1^\perp\rangle \langle \phi_1^\perp|)$$

$$D_1 = \frac{1}{(1 + \cos \theta)} (|\phi_0^\perp\rangle \langle \phi_0^\perp|)$$

$$D_2 = \mathbb{I} - D_0 - D_1,$$

where $|\phi_1^\perp\rangle = (\sin \frac{\theta}{2}|0\rangle + \cos \frac{\theta}{2}|1\rangle)$ and $|\phi_0^\perp\rangle = (\sin \frac{\theta}{2}|0\rangle - \cos \frac{\theta}{2}|1\rangle)$.

This implies that whenever the measurement devices at Alice's side achieve $\Omega = \frac{2 \sin^2 \theta}{(1 + \cos \theta)}$, then it certifies that the measurement operators at Alice's side are the intended POVM devices. In our modified DI proposal, whenever the devices involved in POVMtestAlice do not achieve the value $\Omega = \frac{2 \sin^2 \theta}{(1 + \cos \theta)}$, the protocol aborts. This concludes the proof. □

Finite Sample Analysis in Device Independent QPQ

In recent times, most of the quantum protocols involve sharing of entangled states. In case these are generated by a third party, it is almost mandatory to measure the quantum states used for the protocol to check whether those are actually in the intended form or not. If an entangled state is not what is expected, the adversary may obtain certain extra information, thereby violating the security of the cryptographic scheme. This leads to the development of the idea towards testing the states generated by third party devices before proceeding for the actual protocol. Mayers and Yao first proposed the idea of self testing of quantum devices [81]. For quantum cryptographic protocols, such self testing is defined in DI paradigm that guarantees security under certain assumptions.

Generally the quantum protocols involve sharing of the Bell states or some other two qubit entangled states. For this reason, violation of CHSH inequality [39] or CHSH test [40] is exploited in most of the device independent quantum cryptographic protocols (e.g., [77], [12], [57]). The security analysis generally considers infinite number of samples and asymptotic treatment. However, for all practical purposes, we have finite number of samples and thus we would always like to **minimize** the amount of samples required. In this direction, we study the very recently proposed DI QPQ [77] (a modification of [117] to obtain device independence) as a framework in comparing the number of samples using different games. Thus, here we consider how to use quantum multi party pseudo telepathy game [28] in such scenario and compare its performance with CHSH game in terms of number of samples.

While investigating the performance of CHSH as well as three party pseudo telepathy games for DI-QPQ, it is noted that for a significant range of parameters, the success probability of the pseudo telepathy game is higher than CHSH. The relation between the required sample size and corresponding success probability for testing DI is well known [16] where one can see that the sample size is inversely proportional with the success probability of DI testing. Thus, for a considerable range of parameters, where the success probability of three party pseudo telepathy game is higher compared to CHSH, one can use the first one instead of the second to obtain better efficiency. With this understanding, we propose a certain strategies for testing device independence to minimize the overall sample size.

In Section 6.1, we provide a brief background related to our work before delving into the detailed explanation of our contributions in Section 6.2. Next in Section 6.3, we present our first strategy for certifying the states used in [117]. This strategy involves the server applying a simple unitary operation (CNOT) on the qubits to transform the states into a unitary equivalent of EPR pairs, thereby reducing the required sample size for testing. Furthermore, in Section 6.4, we introduce another strategy where the server transforms the original states from [117] into three-qubit entangled states and certifies them using a three-party pseudo-telepathy game. This alternative approach aims to further minimize the overall sample size required for testing in finite sample scenarios.

6.1 Background

In this section we present several related backgrounds.

6.1.1 CHSH and Parity Game

The CHSH game [40] is played by two players, Alice and Bob (in the same team) are not allowed to communicate in any manner after the initial setup where they may share an entangled state. The referee provides one random bit x to Alice and one random bit y to Bob. Alice has to provide the referee a bit a and Bob has to send b . The referee declares Alice and Bob the winner if $a \oplus b = x \wedge y$; otherwise they are considered defeated.

When Alice and Bob participate in classical set-up, the maximum success probability they can achieve is 0.75. However, when they share each particle of a maximally entangled state and follow some specific kind of measurement strategy, they can win the game with the probability $\cos^2(\frac{\pi}{8})$. Instead of exploiting the maximally entangled state $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$, if Alice and Bob share any (non-maximally entangled) state then the success probability reduces. Such states have been exploited in [77].

In the parity (also known as multi-party pseudo telepathy) game [28], each player A_i is given an input bit x_i and must generate an output bit y_i . The players are guaranteed that the total number of 1's in their inputs is even. Without communication after receiving their inputs, the players are tasked with producing a collective output that contains an even number of 1's if and only if the input contains a multiple of 4 number of 1's. More formally, it requires that $\sum_i^n y_i \equiv \frac{1}{2} \sum_i^n x_i \pmod{2}$, provided $\sum_i^n x_i \equiv 0 \pmod{2}$. If we consider the game for three parties, then the maximum success probability achieved in classical case equals to 0.75. However, if the three parties share GHZ state and perform some particular measurements, they can achieve success with certainty (probability 1) in the quantum case. To match it with the ideas in [77], instead of the GHZ state $\frac{1}{\sqrt{2}}(|000\rangle + |111\rangle)$, if three parties share any other entangled state, the maximum success probability decreases. However, still this will be significantly better for certain range of parameters than that of [77]. This is explained in Section 6.4.

6.1.2 Estimation of Sample Size For Finite Sample Scenario

Generally if we like to distinguish one event having probability p and another having probability $p(1+\epsilon)$, where ϵ is small, then the approximate number of samples required is $O(\frac{1}{p\epsilon^2})$. Informally speaking, one may have a confidence of more than 99% in distinguishing two events with $\frac{64}{p\epsilon^2}$ samples. A more involved expression related to sample size in finite sample scenario can be obtained using Chernoff-Hoeffding [59] bound which is already mentioned in Chapter 4 Proposition 1.

In our case, if the test succeeds, we set $X_i = 1$; otherwise $X_i = 0$. Let us consider $\mathbb{E}[X] = \mathbb{E}[X_i] = p$ and let the variable X denotes the actual success probability p' . Now the question is how large should “the number of samples” be so that we get a good “accuracy” with high “confidence”? More precisely, suppose we want to estimate the success probability p within an error margin of ϵp and confidence $1 - \gamma$, that is,

$$\Pr[|p' - p| \leq \epsilon p] \geq 1 - \gamma, \quad (6.1)$$

where p' and p are the estimated and the expected values respectively. Comparing Equation (6.1) with Proposition 1, and given ϵ , p and γ , we obtain $\exp(-2\epsilon^2 p^2 m) \leq \gamma$, i.e., $m \geq \frac{1}{2\epsilon^2 p^2} \ln \frac{1}{\gamma}$. This implies that as the value of the success probability increases, the required sample size decreases. Denoting the maximum success probability for a specific θ by p_{max} , one can write,

$$m_{opt} = \frac{1}{2\epsilon^2 p_{max}^2} \ln \frac{1}{\gamma}. \quad (6.2)$$

This m_{opt} gives the optimal value of the sample size required to certify a given state where the value of θ corresponding to this state is already known.

6.1.3 Device Independence in QPQ

Here we are interested in investigating how the number of samples towards testing an entangled state can be reduced. Thus, instead of getting into tedious security proofs based on several complicated assumptions, we like to present our assumptions related to device independence. We consider that the required qubits, the quantum gates (unitary operations) and the measurement devices will be provided by the third party. That is, in the DI setting, the security of the protocol can be guaranteed even after removing this trustful assumption over the source, circuits and measurement devices. In the DI-QPQ protocol, the server asks for non-optimally entangled states from a third party and also the measurement devices are purchased from outside. The claimed idea of [77] is as follows.

The two-qubit entangled state involved in Quantum Private Query (QPQ) protocol is of the form

$$|\psi_{QPQ}\rangle = \frac{1}{\sqrt{2}}(|0\rangle_B |\phi_0\rangle_A + |1\rangle_B |\phi_1\rangle_A), \quad (6.3)$$

where $|\phi_0\rangle_A = \cos(\frac{\theta}{2})|0\rangle + \sin(\frac{\theta}{2})|1\rangle$ and $|\phi_1\rangle_A = \cos(\frac{\theta}{2})|0\rangle - \sin(\frac{\theta}{2})|1\rangle$. The success probability of this version of CHSH game (this is not exactly the CHSH game with

maximally entangled state) for this state $|\psi_{QPQ}\rangle$ will be $\frac{1}{8}(\sin\theta(\sin\psi_1 + \sin\psi_2) + \cos\psi_1 - \cos\psi_2) + \frac{1}{2}$ where $|\psi_1\rangle$ and $|\psi_2\rangle$ are the chosen measurement basis and this success probability value can be maximized by choosing appropriate measurement basis $|\psi_1\rangle$ and $|\psi_2\rangle$ for a particular θ .

From the expression derived in section 6.1.2, it is clear that the expected sample size is inversely proportional with the success probability. So, when we consider the finite sample device independent QPQ protocol, we have to maximize the success probability corresponding to a particular state (i.e., for a particular value of θ) to optimize the overall sample size. This is done by properly choosing the values of ψ_1, ψ_2 . Note that this optimal choice of ψ_1 and ψ_2 is only valid for the purpose of DI testing as this ψ_1 and ψ_2 is not involved in the actual execution of QPQ protocol [77]. However for testing purposes, it is better to use the optimized basis for lesser number of samples.

In the DI-QPQ protocol [77], Bob and Alice share entangled states of the form $\frac{1}{\sqrt{2}}(|0\rangle_B|\phi_0\rangle_A + |1\rangle_B|\phi_1\rangle_A)$, where $|\phi_0\rangle_A = \cos(\frac{\theta}{2})|0\rangle + \sin(\frac{\theta}{2})|1\rangle$ and $|\phi_1\rangle_A = \cos(\frac{\theta}{2})|0\rangle - \sin(\frac{\theta}{2})|1\rangle$. The value of θ is known to all. Bob chooses two measurement bases namely $\{|\psi_1\rangle, |\psi_1^\perp\rangle\}$ and $\{|\psi_2\rangle, |\psi_2^\perp\rangle\}$, to play the CHSH game locally. Here, $|\psi_1\rangle = \cos\frac{\psi_1}{2}|0\rangle + \sin\frac{\psi_1}{2}|1\rangle$ and $|\psi_2\rangle = \cos\frac{\psi_2}{2}|0\rangle + \sin\frac{\psi_2}{2}|1\rangle$.

Thus, Bob gets the success probability in terms of θ, ψ_1 and ψ_2 which is equal to $\frac{1}{8}(\sin\theta(\sin\psi_1 + \sin\psi_2) + \cos\psi_1 - \cos\psi_2) + \frac{1}{2}$. To maximize the quantity, we have to maximize $\sin\theta(\sin\psi_1 + \sin\psi_2) + \cos\psi_1 - \cos\psi_2$. Calculation shows that the optimal value of ψ_1, ψ_2 corresponding to a particular θ will be $\psi_1 = (\frac{\pi}{2} - \tan^{-1}(\text{cosec } \theta))$ and $\psi_2 = (\frac{\pi}{2} + \tan^{-1}(\text{cosec } \theta))$. So, the optimal sample size required to test the source device in two party scenario can be found by the expression 6.2 where the value of ψ_1, ψ_2 corresponding to the value p_{max} will be $\psi_1 = (\frac{\pi}{2} - \tan^{-1}(\text{cosec } \theta))$ and $\psi_2 = (\frac{\pi}{2} + \tan^{-1}(\text{cosec } \theta))$. While evaluating with our new proposal, we will compare with this optimized data and show when we can obtain better results.

A Caveat on Device Independence and Security Proofs

Now it is important to describe what provides the Device Independence in [77]. The proof of device independence is varied and not streamlined. In [77], the claim of device independence comes from the following.

- The server (Bob) asks for entangled states of the form $\frac{1}{\sqrt{2}}(|0\rangle_B|\phi_0\rangle_A + |1\rangle_B|\phi_1\rangle_A)$ from the third party as described before. This is basically dependent on θ , i.e., the server provides the value of θ to the third party and the third party provides the required (non-maximal) entangled states.
- The server obtains the measurement devices from the third party too, that will be able to measure in certain measurement basis. These measurement devices are memoryless, and thus each measurement will be independent. Further during the run time, it is assumed that the measurement devices cannot communicate to anybody other than Bob i.e., no information is leaked from the devices.

Based on these assumptions, it is claimed that by performing the CHSH test Bob should obtain certain results related to success probability which he already knows. In case the experimental data closely matches with what he expects, then he will believe on the entangled states obtained as well as the measurement devices which were provided by the third party.

We would like to add the following point here. When the server (Bob) receives an entangled state as above, he may keep one particle with him and communicate the other one to the client (Alice). This is because the idea of Device Independence exploits non-locality. With one measurement device at Bob's side and another at Alice's, the security notions should work if they play the game and then publicly announce the classical outcome. Then Bob and Alice will get to know each other's input as well as outcome after completion of the game and consequently together can estimate whether the correct state is supplied. On the other hand there could be an argument that Alice may be colluding with the third party and possibly that is the reason the complete game was played in the server side for checking the states in [77]. However, the exact security issues here are not clear. On the other hand, this does not affect the work in this initiative as we are primarily interested about studying the number of samples and not the security issues.

We conclude this discussion with some issues related to security proofs. In the domain of cryptology, there are two directions.

- One may provide certain schemes with design details as well as certain justifications towards security and then wait for the cryptanalytic results. This mostly happens in the actual implementations that are in the application domain. The cryptanalytic efforts continue and once a system is attacked, necessary countermeasures are taken. However, no specific formal security proof is provided. For example, design of commercial stream or block ciphers still follow this line. This was the scenario when BB84 protocol [21] was first proposed as, at that time, the security claims were justified from certain laws of Physics.
- Providing schemes with complete security proofs. In this case certain basic assumptions are considered and based on that there are formal-looking security proofs. These are mostly popular in theoretical world. However, certain systems are arriving in market where security proofs are advertised. The main problem in this domain is that in certain cases flaws are identified in many security proofs. In fact, larger the proof, lesser the confidence as many of the long proofs require more serious attention. However, in the positive direction we must appreciate that after the publication of the BB84 protocol, in last three decades researchers have noted many important theoretical proofs justifying several security aspects of BB84 and its variants.

This is an age-old philosophical debate. In this chapter, the DI idea that we mention (towards reducing the number of samples) using Pseudo-Telepathy is not supported by rigorous proof. However, one may refer to [78, 89] and the references therein to get a view of how pseudo-telepathy games may yield device-independent certification given an entangled state.

6.2 Contribution of this chapter

In this chapter, we propose various strategies to certify the entangled states utilized in the protocol presented in [117] with the aim of reducing the required sample size compared to the test described in [77] for finite sample scenario. Our research in this direction yields two key results, which can be summarized as follows.

- In Section 6.3, we note that the test for device independence should be applied on a slightly modified state than the state being used as in [77]. This provides a much better probability compared to that has been achieved in [77], with the expense of one additional CNOT gate only. In fact this shows that how even without considering the maximally entangled state, one can simulate the CHSH game like behaviour by changing the measurement basis in one measurement device.
- In Section 6.4, we exploit the three-party Pseudo Telepathy game for a transformed three-qubit non maximally entangled state and show how it provides even better probability.

6.3 Analysis of CHSH game with modified two-qubit entangled states

In this section, we analyze case by case situation of the CHSH test for a modified two-qubit entangled state of the form

$$\frac{1}{\sqrt{2}}(\cos \frac{\theta}{2}|00\rangle + \sin \frac{\theta}{2}|01\rangle + \cos \frac{\theta}{2}|11\rangle - \sin \frac{\theta}{2}|10\rangle). \quad (6.4)$$

The motivation here is as follows. In the QPQ protocol [117], generally the client learns only a few bits of the shared secret key, while the server learns it all. This is done by certain modification of a quantum key distribution protocol. The entangled state of equation (6.3), used in [117], could provide expected $\frac{1}{2} \sin^2 \theta$ proportion of shared secret key bits to the client. Generally, the client will try to learn only a few bits and thus the value of θ will be very small. The method presented in [77] requires lower probability (more samples) for small θ . We show that with proper choice of the entangled state this can be improved a lot. In fact, one may keep the DI-QPQ protocol [77] exactly the same, but use our strategy only for testing DI.

6.3.1 Success probability calculation

In this case, Bob performs CNOT operation over the original two qubit state shared in DI-QPQ protocol [77] by considering the first qubit of the state as a control bit and second qubit as a target bit. The resulting state after performing this operation will be of the form as mentioned in equation 6.4. We have already mentioned the details of the game in Section 6.1.1.

1. **For input $xy = 00$:** Bob's first quantum device measures the first qubit of the modified state in $\{|0\rangle, |1\rangle\}$ basis and the second quantum device measures the second qubit of the modified state in $\{|\psi_1\rangle, |\psi_1^\perp\rangle\}$ basis. In this case, the probability of obtaining each of 00 and 11 from the two quantum devices (as output) is $\frac{1}{2} \cos^2(\frac{\theta-\psi_1}{2})$ and $\frac{1}{2} \cos^2(\frac{\theta-\psi_1}{2})$ respectively. So, the total winning probability in this case is $\cos^2(\frac{\theta-\psi_1}{2})$.
2. **For input $xy = 01$:** Bob's first quantum device measures the first qubit of the modified state in $\{|0\rangle, |1\rangle\}$ basis and the second quantum device measures the second qubit of the modified state in $\{|\psi_2\rangle, |\psi_2^\perp\rangle\}$ basis. In this case, the probability of obtaining each of 00 and 11 from the two quantum devices (as output) is $\frac{1}{2} \cos^2(\frac{\theta-\psi_2}{2})$ and $\frac{1}{2} \cos^2(\frac{\theta-\psi_2}{2})$ respectively. So, the total winning probability in this case is $\cos^2(\frac{\theta-\psi_2}{2})$.
3. **For input $xy = 10$:** Bob's first quantum device measures the first qubit of the modified state in $\{|+\rangle, |-\rangle\}$ basis and the second quantum device measures the second qubit of the modified state in $\{|\psi_1\rangle, |\psi_1^\perp\rangle\}$ basis. In this case, the probability of obtaining each of 00 and 11 from the two quantum devices (as output) is $\frac{1}{4}[\cos(\frac{\theta-\psi_1}{2}) - \sin(\frac{\theta-\psi_1}{2})]^2$ and $\frac{1}{4}[\cos(\frac{\theta-\psi_1}{2}) - \sin(\frac{\theta-\psi_1}{2})]^2$ respectively. So, the total winning probability in this case is $\frac{1}{2}[\cos(\frac{\theta-\psi_1}{2}) - \sin(\frac{\theta-\psi_1}{2})]^2$.
4. **For input $xy = 11$:** Bob's first quantum device measures the first qubit of the modified state in $\{|+\rangle, |-\rangle\}$ basis and the second quantum device measures the second qubit of the modified state in $\{|\psi_2\rangle, |\psi_2^\perp\rangle\}$ basis. In this case, the probability of obtaining each of 01 and 10 from the two quantum devices (as output) is $\frac{1}{4}[\cos(\frac{\theta-\psi_2}{2}) + \sin(\frac{\theta-\psi_2}{2})]^2$ and $\frac{1}{4}[\cos(\frac{\theta-\psi_2}{2}) + \sin(\frac{\theta-\psi_2}{2})]^2$ respectively. So, the total winning probability in this case is $\frac{1}{2}[\cos(\frac{\theta-\psi_2}{2}) + \sin(\frac{\theta-\psi_2}{2})]^2$.

As all the cases can happen with equal probability (for random choice of inputs), the overall probability of winning the CHSH game with this modified two-qubit entangled state is

$$\frac{1}{2} + \frac{1}{8}[\cos(\theta - \psi_1) + \cos(\theta - \psi_2) - \sin(\theta - \psi_1) + \sin(\theta - \psi_2)].$$

6.3.2 Appropriate choice of measurement basis

From the discussion of the previous subsection, we can see that for the modified two-qubit entangled state, Bob gets the success probability in terms of θ , ψ_1 and ψ_2 which is equal to $\frac{1}{2} + \frac{1}{8}[\cos(\theta - \psi_1) + \cos(\theta - \psi_2) - \sin(\theta - \psi_1) + \sin(\theta - \psi_2)]$. To maximize the quantity, we have to maximize $\cos(\theta - \psi_1) + \cos(\theta - \psi_2) - \sin(\theta - \psi_1) + \sin(\theta - \psi_2)$.

Now, we can write,

$$[\cos(\theta - \psi_1) - \sin(\theta - \psi_1)] + [\cos(\theta - \psi_2) + \sin(\theta - \psi_2)]$$

Setting $\theta - \psi_1 = A$, $\theta - \psi_2 = B$, $1 = r_1 \sin \phi_1 = r_1 \cos \phi_1$ (for the first half of the expression) and $1 = r_2 \sin \phi_2 = r_2 \cos \phi_2$ (for the second half of the expression), we

get

$$\begin{aligned}
& (r_1 \sin \phi_1 \cos A - r_1 \cos \phi_1 \sin A) \\
& + (r_2 \sin \phi_2 \cos B + r_2 \cos \phi_2 \sin B) \\
= & r_1 \sin(\phi_1 - A) + r_2 \sin(\phi_2 + B),
\end{aligned}$$

where $r_1^2 = r_2^2 = 2$ and $\tan \phi_1 = \tan \phi_2 = 1$ i.e., $\phi_1 = \phi_2 = \tan^{-1}(1) = \frac{\pi}{4}$.

Again, the value $r_1 \sin(\phi_1 - A) + r_2 \sin(\phi_2 + B)$ will be maximum when both $\sin(\phi_1 - A) = 1$ and $\sin(\phi_2 + B) = 1$ i.e., when $(\phi_1 - A) = \frac{\pi}{2}$ and $(\phi_2 + B) = \frac{\pi}{2}$. From that, after putting the value of A and B we get, $\psi_1 = (\frac{\pi}{4} + \theta)$ and $\psi_2 = (\theta - \frac{\pi}{4})$.

From the discussion, it is clear that the optimal value of $|\psi_1\rangle$ and $|\psi_2\rangle$ corresponding to a particular θ will be $\psi_1 = (\frac{\pi}{4} + \theta)$ and $\psi_2 = (\theta - \frac{\pi}{4})$. So, the success probability corresponding to each theta will be maximum for this particular choice of measurement basis. By putting this value into the success probability expression of the modified state (as derived in previous subsection), we can see that for this particular choice of measurement basis, the success probability value of CHSH game with this modified state for different values of θ is constant and this success probability value is the maximum success probability that we can get for two-qubit entangled states in CHSH game. This is indeed natural as we are making local transformation at one side and then accordingly modifying the measurement basis.

We like to refer that this success probability is significantly greater than what could be obtained in [77] for $\theta \leq \frac{\pi}{2}$ that is presented in Figure 6-2.

6.4 Analysis of three-party quantum pseudo telepathy with transformed three-qubit entangled states

In this section, we analyze case by case situation of the proposed multi party pseudo telepathy (parity) test for a three-qubit entangled states of the form

$$\frac{1}{\sqrt{2}}(\cos \frac{\theta}{2}|000\rangle + \sin \frac{\theta}{2}|010\rangle + \cos \frac{\theta}{2}|111\rangle - \sin \frac{\theta}{2}|100\rangle).$$

We have already mentioned the details of the game in Section 6.1.1.

1. **For input $x_1x_2x_3 = 000$:** The quantum devices perform Hadamard operation over individual qubits and measure each qubit in $\{|0\rangle, |1\rangle\}$ basis. In this case, probability of obtaining each of 000, 110, 011, 101 from the three quantum devices (as output) are $\frac{1}{4} \cos^2(\frac{\theta}{2})$, $\frac{1}{4} \cos^2(\frac{\theta}{2})$, $\frac{1}{4}(\cos \frac{\theta}{2} - \sin \frac{\theta}{2})^2$ and $\frac{1}{4}(\cos \frac{\theta}{2} + \sin \frac{\theta}{2})^2$ respectively. So, the total winning probability in this case is $\frac{1}{4}(3 + \cos \theta)$.
2. **For input $x_1x_2x_3 = 110$:** Each of the first two quantum devices perform the unitary operator S (as described in [28]) over the first two particles. Then all the devices performs Hadamard operation over the individual qubits and measure each qubit in $\{|0\rangle, |1\rangle\}$ basis. In this case, probability of getting each of 100, 010, 001, 111 from the three quantum devices (as output) are $\frac{1}{4}$, $\frac{1}{4}$, $\frac{1}{4} \cos^2(\frac{\theta}{2})$ and

$\frac{1}{4} \cos^2(\frac{\theta}{2})$ respectively. Thus, the total winning probability in this case becomes $\frac{1}{4}(3 + \cos \theta)$.

3. **For input $x_1x_2x_3 = 011$:** The devices first perform S over the last two qubits and then all the devices apply Hadamard operation over the individual qubits and then measure each qubit in $\{|0\rangle, |1\rangle\}$ basis. In this case, probability of getting each of 100, 010, 001, 111 from the three quantum devices (as output) are $\frac{1}{16}[3 + \cos \theta + 2 \sin \theta]$, $\frac{1}{16}[3 + \cos \theta - 2 \sin \theta]$, $\frac{1}{16}[3 + \cos \theta - 2 \sin \theta]$ and $\frac{1}{16}[3 + \cos \theta + 2 \sin \theta]$ respectively. Hence, here we obtain the total winning probability as $\frac{1}{4}(3 + \cos \theta)$.
4. **For input $x_1x_2x_3 = 101$:** The devices first perform S over the first and third qubits and then all the devices perform Hadamard operation over individual qubits and measure each qubit in $\{|0\rangle, |1\rangle\}$ basis. In this case, probability of getting each of 100, 010, 001, 111 from the three quantum devices (as output) are $\frac{1}{16}[3 + \cos \theta + 2 \sin \theta]$, $\frac{1}{16}[3 + \cos \theta - 2 \sin \theta]$, $\frac{1}{16}[3 + \cos \theta + 2 \sin \theta]$ and $\frac{1}{16}[3 + \cos \theta - 2 \sin \theta]$ respectively. Thus the winning probability becomes $\frac{1}{4}(3 + \cos \theta)$.

As all the cases can happen with equal probability (for random choice of inputs from the set $\{000, 110, 011, 101\}$), the overall probability of winning the multi party pseudo telepathy game with this specified form of three qubit entangled state is

$$4 \times \frac{1}{4} \times \frac{1}{4}(3 + \cos \theta) = \frac{1}{4}(3 + \cos \theta)$$

which is equal to 1 (i.e., maximum) when $\theta = 0$, i.e., the success probability will be maximum for three qubit maximally entangled (GHZ) states. We like to refer that this success probability is greater than what could be obtained in [77] for certain ranges of θ that is presented in Figure 6-2.

6.4.1 Transformation of two-qubit state into three-qubit

In the DI-QPQ [77] set-up, Bob holds the initial two qubit entangled state, and say, that it can perform either CHSH test or parity test locally before proceeding for the actual QPQ protocol. When Bob performs the parity test locally, he has to first transform the initial two qubit entangled state $|\psi_{QPQ}\rangle$ into three qubit entangled state $|\psi_{3QPQ}\rangle$ as follows.

- Bob first performs the CNOT operation over the initial two qubit entangled state by considering first qubit as a control bit and second qubit as a target bit.
- After performing the CNOT operation, Bob will add an ancilla qubit $|0\rangle$ in his end and perform Toffoli operation by considering the two qubits of the modified entangled state as control bit and the ancilla qubit as a target bit.
- After performing these operations, the resulting state will be of the form

$$\frac{1}{\sqrt{2}}(\cos \frac{\theta}{2}|000\rangle + \sin \frac{\theta}{2}|010\rangle + \cos \frac{\theta}{2}|111\rangle - \sin \frac{\theta}{2}|100\rangle)$$

The circuit diagram corresponding to this transformation is shown in figure 6-1.

- Bob will perform multiparty pseudo telepathy (parity) game [28] with this transformed state.

Now the success probability of the parity game with this transformed three qubit state will be $\frac{1}{4}(3 + \cos \theta)$ which equals 1 for $\theta = 0$.

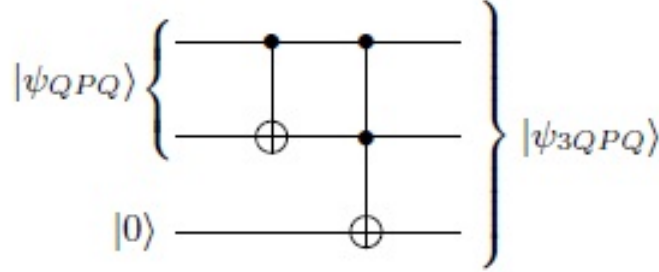


Figure 6-1: Circuit diagram for transformed state

6.4.2 Comparative study

Let us consider the actual two-qubit entangled state shared in QPQ protocol which is of the form $\frac{1}{\sqrt{2}}(\cos \frac{\theta}{2}|00\rangle + \sin \frac{\theta}{2}|01\rangle + \cos \frac{\theta}{2}|10\rangle - \sin \frac{\theta}{2}|11\rangle)$, then the success probability of CHSH game (maximum success probability corresponding to each θ) for this state equals to $\frac{1}{8}(\sin \theta(\sin \psi_1 + \sin \psi_2) + \cos \psi_1 - \cos \psi_2) + \frac{1}{2}$ where $\psi_1 = (\frac{\pi}{2} - \tan^{-1}(\text{cosec } \theta))$ and $\psi_2 = (\frac{\pi}{2} + \tan^{-1}(\text{cosec } \theta))$.

Instead of the actual state, if we consider the modified two-qubit entangled state of the form $\frac{1}{\sqrt{2}}(\cos \frac{\theta}{2}|00\rangle + \sin \frac{\theta}{2}|01\rangle + \cos \frac{\theta}{2}|11\rangle - \sin \frac{\theta}{2}|10\rangle)$, then according to the discussion in section 6.3, the success probability of CHSH game (maximum success probability corresponding to each θ) for this state equals to $\frac{1}{2} + \frac{1}{8}[\cos(\theta - \psi_1) + \cos(\theta - \psi_2) - \sin(\theta - \psi_1) + \sin(\theta - \psi_2)]$ where $\psi_1 = (\theta + \frac{\pi}{4})$ and $\psi_2 = (\theta - \frac{\pi}{4})$. With this particular choice of basis the actual success probability is further improved and it provides the same result as obtained in the CHSH game with maximally entangled state.

Further, if we consider the transformed three-qubit entangled state of the form $\frac{1}{\sqrt{2}}(\cos \frac{\theta}{2}|000\rangle + \sin \frac{\theta}{2}|010\rangle + \cos \frac{\theta}{2}|111\rangle - \sin \frac{\theta}{2}|100\rangle)$, then according to the discussion in section 6.4, the success probability of parity game for this state equals to $\frac{1}{4}(3 + \cos \theta)$

The comparative study between the success probability values of two games (for different forms of states) corresponding to different values of θ from 0 to $\frac{\pi}{2}$ is shown in figure 6-2.

From the graph, it is clear that for CHSH game, the value of success probability varies between 0.75 to $\cos^2 \frac{\pi}{8}$ for the actual state shared in QPQ protocol and the

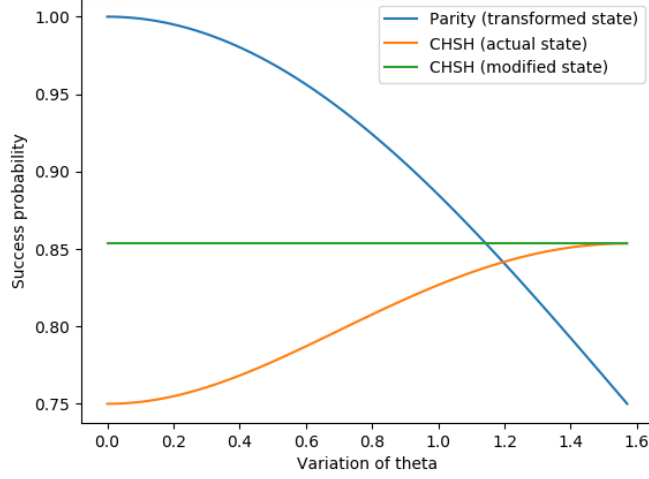


Figure 6-2: Comparative study of success probabilities between CHSH and parity game for DI-QPQ protocol

success probability of the two qubit modified entangled state (as discussed in section 6.4) remains constant i.e., $\cos^2 \frac{\pi}{8}$ irrespective of the value of θ . For the parity game, the value of the success probability for the transformed three qubit entangled state (as discussed in section 6.4) varies between 1 to 0.75. From the graph (as well as from calculation), it is clear that at $\theta \approx 1.14$, the success probability of parity game and the success probability of CHSH game for the modified two qubit state becomes equal. Thus, for all the values of $\theta < 1.14$, the success probability of parity game for transformed three qubit state is higher compared to the success probability of CHSH game for the modified two qubit state. On the other hand, for $\theta \geq 1.14$, the success probability of CHSH game for modified two qubit state is higher compared to the success probability of parity game for transformed three qubit state.

Similarly, for the value of $\theta \approx 1.2$, the success probability of parity game and the success probability of CHSH game for the actual two qubit state becomes equal and beyond that point, the success probability of CHSH game for actual two qubit state is higher compared to the success probability of parity game for transformed three qubit state. However, the success probability value of CHSH game for the modified two qubit state is always higher as compared to the success probability value for the actual two qubit state and the two become equal for $\theta = 1.57$.

In case we are interested for small values of θ , the parity game as in Section 6.4 will be the best suited for testing DI. Thus [77, Algorithm 1] should be parameterised based on the value of θ . Further parity game does not require modifying the measurement bases as it is required for the CHSH test as described in Section 6.3.

6.4.3 Towards security analysis for finite samples

As we consider finite number of samples in our modified testing mechanism, in testing phase, we need to check whether the success probability value lies within the interval $[p_{QPQ} - \epsilon p_{QPQ}, p_{QPQ} + \epsilon p_{QPQ}]$, where p_{QPQ} is the intended success probability corresponding to a particular form of state (i.e., for a particular value of θ) and ϵ is the accuracy parameter chosen by the server (Bob). When the states successfully pass this test, Bob proceeds further for the actual QPQ protocol, otherwise he aborts.

In [77], the authors outlined an attack strategy over the QPQ protocol where they have shown that if there is ϵ_A amount of bias in the choice of measurement basis by the client (i.e., Alice), then she can extract $(\frac{1}{2} + 2\epsilon_A^2) \sin^2 \theta$ fraction of entire key stream, where the amount of extra information leaked is $2\epsilon_A^2 \sin^2 \theta$. Towards resisting such leakage (which arises due to the finite sample size), Bob must bound the value of ϵ_A so that the additional information which is leaked to Alice should be infinitesimally small. In this direction, one may quantify the security of a protocol in the following manner.

The additional information leaked to the adversary (client) for our optimal sample protocol due to the biased choice of the client's measurement basis will be proportional to the value of ϵ , where ϵ is the accuracy parameter chosen by the server. This can be justified as follows. Let, instead of the correct states, Bob is provided with the states of the form $(\alpha|0\rangle_B|\phi_0\rangle_A + \beta|1\rangle_B|\phi_1\rangle_A)$ where $|\alpha|^2 = (\frac{1}{2} + \epsilon_A)$ and $|\beta|^2 = (\frac{1}{2} - \epsilon_A)$.

When Bob performs the CHSH test, the success probability for the modified states become $p' = \frac{1}{2} + \frac{1}{8} \sin \theta (\sin \psi_1 + \sin \psi_2) + \frac{1}{4} \sqrt{\frac{1}{4} - \epsilon_A^2} (\cos \psi_1 - \cos \psi_2) + \frac{1}{4} \epsilon_A \cos \theta (\cos \psi_1 + \cos \psi_2)$. Now p' must lie within the interval $[p_{QPQ} - \epsilon p_{QPQ}, p_{QPQ} + \epsilon p_{QPQ}]$, where p_{QPQ} is the intended success probability of the modified state and ϵ is the accuracy parameter chosen by Bob.

Thus from the lower and upper bounds, we get $\epsilon_A^2 \geq -\frac{2\epsilon p_{QPQ}}{\cos \psi_1}$ and $\epsilon_A^2 \leq \frac{2\epsilon p_{QPQ}}{\cos \psi_1}$ respectively. Since negative ϵ_A is not meaningful, we have the solution as

$$\epsilon_A \leq \sqrt{\frac{2\epsilon p_{QPQ}}{\cos \psi_1}}. \quad (6.5)$$

Thus, to deceive Bob, the states should be prepared in such a way that the value of ϵ_A must satisfy the condition $\epsilon_A \leq \sqrt{\frac{2\epsilon p_{QPQ}}{\cos \psi_1}}$. Otherwise, the value of p' will not lie within the specified interval and Bob has to abort the protocol. As for a given θ , the values of p_{QPQ} , ψ_1 and ψ_2 are constant, we can write $\epsilon_A \leq k\sqrt{\epsilon}$, where k is a constant.

Similarly, for the given erroneous state $(\alpha|0\rangle_B|\phi_0\rangle_A + \beta|1\rangle_B|\phi_1\rangle_A)$, when Bob performs the parity test, the success probability of parity test for the transformed states becomes $p'' = \frac{1}{4} [1 + \cos \theta + 2\sqrt{\frac{1}{4} - \epsilon_A^2} (1 + \cos \theta)]$. This value of p'' must lie within the interval $[p_{QPQ} - \epsilon p_{QPQ}, p_{QPQ} + \epsilon p_{QPQ}]$, where p_{QPQ} is the intended success probability of the transformed state.

Now from the left and right inequalities, we get $\epsilon_A^2 \geq -\epsilon$ and $\epsilon_A^2 \leq \epsilon$ respectively.

Since negative ϵ_A is not meaningful, we have the solution as

$$\epsilon_A \leq \sqrt{\epsilon}. \quad (6.6)$$

Analyzing both the relation between ϵ_A and ϵ for CHSH test and parity test in equations (6.5) and (6.6) respectively, one may conclude that the maximum value of ϵ_A is related to the square root of the value of chosen accuracy parameter (i.e., ϵ). From the discussion in [77], the additional information leaked to Alice equals to $2\epsilon_A^2 \sin^2 \theta$. As the value of ϵ_A is proportional with the square root of the chosen accuracy parameter ϵ , the maximum information leaked to Alice will be proportional with the value of the chosen accuracy parameter ϵ .

6.5 Discussion and Conclusion

In this chapter, we propose several strategies to improve the test of device independence in the Device Independent Quantum Private Query Proposal [77]. Our motivation comes from the analysis in finite sample scenario, which is mandatory for actual implementation of the protocol. We derive the relation between the required sample size and corresponding success probability and propose optimal testing mechanisms for DI-QPQ protocol. CHSH tests on different versions of the entangled states are studied. Further, we also consider the three-party Pseudo Telepathy as a tool for testing DI and show that it provides significantly better results for practical purposes.

Proposal For Multi-User Semi Device Independent QPQ

Although the development in the field of QPQ is evident from the significant number of recent proposals, they mostly consider the single-user, single-server scenario. For a more practical multi-user scenario, these single-user schemes need to be executed multiple times between the server and each user, which is inefficient. To overcome this inefficiency, some multi-user schemes have been proposed recently [116, 119, 110]. Ye et al. [119] basically implemented the QPQ scheme [52] repeatedly for each user, which has already been cryptanalyzed in [121]. The other two proposals [116, 110] consider the existence of a semi-trusted server to generate an oblivious key between the users so that they can jointly retrieve items of common interest. However, neither of these solutions is practical because users typically want to retrieve different items from the database without revealing their choices to others (for privacy). For items of common interest, one user may simply retrieve the intended bits from the database and share those bit values with other users using a QKD scheme. This implies that the solutions proposed in [116, 110] are not practical. Additionally, the assumption of a semi-honest party is not realistic in a distrustful scheme.

Here in this chapter, we propose a new multi-user QPQ scheme that overcomes the limitations of the existing schemes. Our proposal allows each user to independently retrieve different data bits without revealing their items of interest. In our proposal, each of the users can retrieve an optimal number of raw key bits during the oblivious key generation phase. Additionally, our scheme offers semi-device independent security by certifying the input states and the measurement devices at each of the client's sides and by putting trustful assumptions over the functionality of the other devices. In contrast to other multi-user proposals that discuss security under certain eavesdropping strategies, here we perform a formal evaluation of the security concerns and are able to determine the maximum likelihood of cheating for both the server and users.

Before explaining our exact contributions in detail in Section 7.2, we briefly discuss the limitations of the existing multi-user QPQ schemes in Section 7.1. Then, we describe our semi-DI proposal for multi-user QPQ in Section 7.3, where we try

to overcome the limitations of the existing multi-user QPQ schemes. Finally, in Section 7.4, we formally evaluate the security concerns of our proposal and determine the maximum likelihood of cheating for both the server and users.

7.1 Limitations of the existing multi-user QPQ proposals

Although the developments in the QPQ domain are evident from the several single-user single-server protocols, when considering the multi-user scenario, existing single-user protocols may require multiple executions for each user, leading to increased communication and resource overhead. Recent advancements have introduced multi-user QPQ schemes [116, 119, 110] to reduce these overheads. The proposal by Ye et al. [119] has already been cryptanalyzed in [121]. Here, we discuss the limitations of [116] and [110] that can be summarized as follows.

- The proposals in [116, 110] assume that the users always have the same items of interest, which they retrieve from the database by trusting each other. However, in practice, different users usually want to retrieve different data bits without revealing their choices to anyone for privacy. In such cases, the existing multi-user schemes are not suitable.
- As assumed in [116, 110], if the users are honest with each other, a simpler approach would be for one user to retrieve the intended bits using a single user QPQ scheme (such as [15]), and then share those bits with other users using a QKD scheme (such as [109]). This approach would then reduce the overhead and quantum resources required.
- The proposals in [116, 110] assume the existence of a semi-trusted third-party quantum server. However, for distrustful primitives like QPQ, the assumption about the honest behavior of any involved party (including a third party) is impractical.
- The proposals presented in [116, 110] generate fixed-length raw key bits at each of the client's sides. However, in the ideal scenario, the server should have the freedom to choose how many key bits each client can know.
- In [116, 110], the security concerns are examined by analyzing certain eavesdropping strategies that can be employed by a dishonest user or server instead of providing a formal analysis.

7.2 Contribution of this chapter

This chapter focuses on the QPQ distrustful primitive in multi-user scenarios. Our main contributions in this chapter can be summarized as follows.

1. In this proposal, we come up with a practical multi-client QPQ scheme using GHZ states where different users may query simultaneously for different items from the database. Our proposal removes the trustworthiness from the source (input state generation) device and the devices that perform measurements at the clients' side exploiting the self-testing of GHZ states (following the procedures mentioned in [82] and [115]) and the self-testing of POVM operators (following the strategy mentioned in [15]). To the best of our knowledge, this proposal is the first of its kind in multi-user QPQ, where different clients can simultaneously retrieve different items of interest from the database.
2. Like the proposal in [15], this scheme also utilizes optimal POVM measurements (in distinguishing two non-orthogonal states) at the clients' side, replacing the traditional projective measurement. This ensures the retrieval of the maximum amount of raw key bits (during the oblivious key generation phase) and, consequently, the optimal number of data bits by the clients (in a single query).
3. Contrary to the existing multi-user QPQ proposals that only consider specific eavesdropping strategies, our proposal undergoes a formal evaluation of its security performance considering the security definitions introduced in [15]. We further determine the maximum likelihood of cheating for both the server and clients in the dishonest scenario.

7.3 Our semi-DI-QPQ proposal

Depending on the functionality, our entire protocol is divided into six phases. Those different phases are described below. Note that this proposal follows all the assumptions mentioned in Chapter 3 Section 3.6. Additionally, this proposal also assumes the following.

- The identity operator \mathbb{I}_2 and the unitary operator U at the server's and each client's side, as well as the projective measurement devices at the server's side, work as intended and are trusted.

Here, we have not incorporated the channel noise in this proposal. Therefore, all the operations mentioned here are assumed to be flawless.

1. Entangled State Supply Phase:

- (a) A third party provides \mathcal{K} (where \mathcal{K} is assumed to be asymptotically large) number of $(n + 1)$ -qubit entangled states to the server.

2. Entangled State Sharing Phase:

- (a) For every j -th state of these \mathcal{K} samples received from the third party, the server performs the following steps.

- The server generates random bit $r_j \in_R \{0, 1\}$ for every j -th state (essentially, these randomly generated bits serve as the initial raw key bits for the server).
- If $r_j = 0$, the server performs identity operator (\mathbb{I}_2) to all the qubits of the j -th state. That means, for honest implementation of this case, the j -th state is supposed to be of the form $\frac{1}{\sqrt{2}} (|0\rangle^{\otimes(n+1)} + |1\rangle^{\otimes(n+1)})$.
- If $r_j = 1$, the server performs the unitary operator U to each qubit of the j -th state where U is of the following form.

$$U = \begin{bmatrix} \cos \theta & \sin \theta \\ \sin \theta & -\cos \theta \end{bmatrix}.$$

That means, for honest implementation of this case, the j -th state is supposed to be of the form $\frac{1}{\sqrt{2}} (|0'\rangle^{\otimes(n+1)} + |1'\rangle^{\otimes(n+1)})$ where $|0'\rangle = (\cos \theta|0\rangle + \sin \theta|1\rangle)$ and $|1'\rangle = (\sin \theta|0\rangle - \cos \theta|1\rangle)$.

- After these operations, the server shares the j -th state with the n users such that the i -th qubit of the state corresponds to the i -th user and the $(n + 1)$ -th qubit corresponds to the server.

3. Entangled State Verification Phase:

In this phase, the server and the clients jointly verify the states the untrusted third party provides in a decentralized way. A verification procedure for the multi-particle GHZ states (provided by the untrusted third party) was demonstrated in [82], which is known as θ -protocol. Here, we adopted a simplified version of the θ -protocol mentioned in [115]. For this verification phase, each of the $(n + 1)$ participants (i.e., the n clients and the server) will act as a verifier in different iterations, choose input bits for all the participants, get the corresponding outcomes from them and checks whether these values match a certain condition. The different steps of this phase can be outlined as follows.

- For every $i \in (n + 1)$, the participant \mathcal{P}_i (i.e., the i -th participant) does the following.
 - The participant \mathcal{P}_i chooses $\frac{\gamma\mathcal{K}}{(n+1)}$ samples randomly from the rest of shared states (that are not already chosen for the testing phase), declares the instances publicly and constructs a set Γ_{GHZ}^i with these chosen instances.
 - For the instances in Γ_{GHZ}^i , the $(n + 1)$ -th participant (i.e., \mathcal{P}_{n+1} or the server) declares his randomly chosen r_j values publicly. For an instance in Γ_{GHZ}^i , if the declared value $r_j = 0$, every participant applies the unitary operator \mathbb{I}_2 to their respective qubits of that instance. Otherwise (i.e., for $r_j = 1$), every participant applies the unitary operator U to their respective qubits of that instance.

- For the instances in Γ_{GHZ}^i , the participant \mathcal{P}_i performs $\text{GHZtest}(\Gamma_{\text{GHZ}}^i, \mathcal{P}_i)$ according to the procedure described in Algorithm 10.
- If the set Γ_{GHZ}^i passes this $\text{GHZtest}(\Gamma_{\text{GHZ}}^i, \mathcal{P}_i)$ then the scheme continues, otherwise the scheme terminates.

Algorithm 10: $\text{GHZtest}(\mathcal{S}, \mathcal{P})$

- For every $k \in \mathcal{S}$, \mathcal{P} acts as a verifier and does the following.
 - (a) \mathcal{P} selects a random $(n + 1)$ -bit string such that the string contains even number of 1's and sends the j -th bit of the string to the j -th participant.
 - (b) If the j -th participant receives the input bit 1, he measures his corresponding particle of the k -th shared state in $\left\{ \frac{1}{\sqrt{2}} (|0\rangle + e^{i\theta_{k_j}} |1\rangle), \frac{1}{\sqrt{2}} (|0\rangle - e^{i\theta_{k_j}} |1\rangle) \right\}$ basis for $\theta_{k_j} = \frac{\pi}{2}$ (i.e., the measurement basis in this case is $\left\{ \frac{1}{\sqrt{2}} (|0\rangle + i|1\rangle), \frac{1}{\sqrt{2}} (|0\rangle - i|1\rangle) \right\}$). If the measurement outcome is $\frac{1}{\sqrt{2}} (|0\rangle + i|1\rangle)$, the j -th participant sends $Y_{k_j} = 0$ to \mathcal{P} , otherwise he sends $Y_{k_j} = 1$ to \mathcal{P} (here θ_{k_j} denotes the measurement angle for the j -th participant corresponding to the k -th shared state and Y_{k_j} denotes the measurement outcome of the j -th participant corresponding to the k -th state).
 - (c) Similarly, if the j -th participant receives the input bit 0, he measures his corresponding particle of the k -th shared state in $\left\{ \frac{1}{\sqrt{2}} (|0\rangle + e^{i\theta_{k_j}} |1\rangle), \frac{1}{\sqrt{2}} (|0\rangle - e^{i\theta_{k_j}} |1\rangle) \right\}$ basis for $\theta_{k_j} = 0$ (i.e., the measurement basis in this case is $\left\{ \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle), \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle) \right\}$). If the measurement outcome is $\frac{1}{\sqrt{2}} (|0\rangle + |1\rangle)$, the j -th participant sends $Y_{k_j} = 0$ to \mathcal{P} , otherwise the j -th participant sends $Y_{k_j} = 1$ to \mathcal{P} .
 - (d) From the known chosen measurement angles θ_{k_j} and the corresponding outcomes Y_{k_j} , \mathcal{P} calculates the values of $\bigoplus_j Y_{k_j}$ and $\frac{1}{\pi} \sum_j \theta_{k_j}$ for the k -th state.
 - (e) If $\bigoplus_j Y_{k_j} = \frac{1}{\pi} \sum_j \theta_{k_j} \pmod{2}$ for the k -th state then the scheme continues, otherwise the scheme terminates.

4. Client's POVM Device Verification Phase:

- (a) After *entangled state verification phase*, the server and the clients move on to this phase with all the rest $(\mathcal{K} - \gamma\mathcal{K})$ shared states, referred to as Γ_{client} .
- (b) For every j -th state in the set Γ_{client} , the server performs the following.
 - If $r_j = 0$, the server measures his qubit (i.e., the $(n + 1)$ -th qubit) of the j -th state in $\{|0\rangle, |1\rangle\}$ basis. Otherwise (i.e., for $r_j = 1$) the server measures his qubit in $\{|0'\rangle, |1'\rangle\}$ basis.

- After the measurement, the server announces $a_j = 0$ whenever the outcome at his side for the j -th shared state is either $|0\rangle$ or $|0'\rangle$.
 - The server announces $a_j = 1$ whenever the outcome at his side is either $|1\rangle$ or $|1'\rangle$.
- (c) From the samples in Γ_{client} , each of the clients then selects $\frac{\gamma'}{n}$ fraction of samples randomly and declares those chosen instances publicly.
- (d) Based on the declaration, the clients construct a set $\Gamma_{\text{client}}^{\text{test}}$ which contains all the instances chosen by each of them.
- (e) For the samples in $\Gamma_{\text{client}}^{\text{test}}$, each of the clients does the following.
- A client first performs $\text{ClientKeyGen}(\Gamma_{\text{client}}^{\text{test}})$ according to the procedure described in Algorithm 11 for the set $\Gamma_{\text{client}}^{\text{test}}$.
 - The same client then performs $\text{ClientPOVMtest}(\Gamma_{\text{client}}^{\text{test}})$ according to the procedure described in Algorithm 12 for the same set $\Gamma_{\text{client}}^{\text{test}}$.

Algorithm 11: ClientKeyGen(\mathcal{S})
<ul style="list-style-type: none"> • For every $j \in \mathcal{S}$, the client performs the following steps. <ul style="list-style-type: none"> (a) If the server declared $a_j = 0$, the client uses the measurement device $P^0 = \{P_0^0, P_1^0, P_2^0\}$ to measure her qubit of the j-th state. (b) Similarly, if the server declared $a_j = 1$, the client uses the measurement device $P^1 = \{P_0^1, P_1^1, P_2^1\}$ to measure her qubit of the j-th state.

5. Shared Key Generation Phase:

- (a) After *client's POVM device verification phase*, the clients continue with the remaining shared states ($|\Gamma_{\text{client}}| - |\Gamma_{\text{client}}^{\text{test}}|$), which they denote as Γ_{Key} .
- (b) For the set Γ_{Key} , each of the clients first performs $\text{ClientKeyGen}(\Gamma_{\text{Key}})$ and then determines the original raw key bits based on her measurement outcomes in the following way.
- For every j -th shared state with $a_j = 0$, if the client gets $P_0^0(P_1^0)$, she concludes the j -th raw key bit as 0(1). If she receives P_2^0 , she ignores it.
 - Similarly, for every j -th shared state with $a_j = 1$, if the client obtains $P_0^1(P_1^1)$, she concludes the j -th raw key bit as 0(1). If she receives P_2^1 , she ignores it.
- (c) The server and the clients then advance to the private query phase with the states in Γ_{Key} . This set contains kN many states, where $k > 1$ and k is exponentially smaller than N , the number of bits in the database.
- (d) The server and the clients then conduct some classical post-processing in the next phase using the kN raw key bits received from these shared states.

Algorithm 12: ClientPOVMtest(\mathcal{S})

- The client first separates her instances having the declared value $a_j = 0$ into the set \mathcal{S}^0 , and the rest (where the server declared $a_j = 1$) into the set \mathcal{S}^1 .
- The client assumes that for each set \mathcal{S}^{a_j} (for the declared a_j values by the server), the states at her side are either $\rho_{r_j}^{a_j}$ or $\rho_{r_j \oplus 1}^{a_j}$ (for the j -th raw key bit r_j chosen by the server).
- For each set, the client calculates the parameter Ω^{a_j} as

$$\Omega^{a_j} = \sum_{b, a_j \in \{0,1\}} (-1)^{b \oplus a_j} \text{Tr}[P_b^{a_j} \rho_{r_j}^{a_j}],$$

where $P_b^{a_j}$ is the measurement outcome at the client's side in ClientKeyGen().

- If for every \mathcal{S}^{a_j} ($a_j \in \{0,1\}$),

$$\Omega^{a_j} = \frac{2 \sin^2 \theta}{(1 + \cos \theta)}$$

then the client continues with the scheme, otherwise she aborts the scheme.

6. Private Query Phase:

- (a) The server first divides the entire raw key into N partitions, each with k bits, and announces the positions. Each of them then XOR each of their substrings bitwise to generate N bits long final key. If any of the clients are unaware of the final key bits, they will again be involved in this shared key generation process with another set of clients.
- (b) If none of the clients know any of the final key bits, repeat the scheme.
- (c) The client, who recognizes only the j -th bit of the server's final key F , requests the k -th bit of the database m_k by announcing a permutation P_A that moves the j -th final key bit to the k -th position. The server then applies P_A on his final key F , uses it to encrypt the database using a one-time pad, and sends the encrypted database to the corresponding client, who decrypts and recovers m_k . This way, a client must announce the permutation l times to retrieve l many data bits.
- (d) If a client knows multiple final key bits, she can retrieve multiple intended database bits in a query by announcing the permutation P_A accordingly.

An Example (For Two Users) Considering Honest Implementation

- Suppose for a N -bit database, the server receives $2N$ number of 3-qubit GHZ states of the form $\frac{1}{\sqrt{2}}(|000\rangle + |111\rangle)$ from a third party.

- For every i -th state, the server and the two users do the following.
 - The server randomly chooses r_i , the raw key bit corresponding to the i -th instance (i.e., $r_i \in_R \{0, 1\}$).
 - If $r_i = 0$, the server applies the identity operator \mathbb{I}_2 to every qubit of the i -th state, and if $r_i = 1$, the server applies the unitary operator U (as mentioned in entangled state sharing phase of our proposal) to every qubit of the i -th state.
 - After these operations, the server shares the i -th state with the two users such that the first two qubits of the state belong to the two users and the 3rd qubit belongs to the server.
(i.e., for $r_i = 0$, the server and the two users share the states of the form $\frac{1}{\sqrt{2}}(|000\rangle + |111\rangle)$ and for $r_i = 1$, they share the states of the form $\frac{1}{\sqrt{2}}(|0'0'0'\rangle + |1'1'1'\rangle)$ where $|0'\rangle = \cos\theta|0\rangle + \sin\theta|1\rangle$ and $|1'\rangle = \sin\theta|0\rangle - \cos\theta|1\rangle$)
 - For $r_i = 0$, the server measures his qubit (i.e., the third qubit) of the i -th state in $\{|0\rangle, |1\rangle\}$ basis and for $r_i = 1$, he measures his qubit of the i -th state in $\{|0'\rangle, |1'\rangle\}$ basis.
 - For the i -th state, the server declares a bit $a_i = 0(a_i = 1)$ whenever his measurement outcome is either $|0\rangle(|1\rangle)$ or $|0'\rangle(|1'\rangle)$.
 - For $a_i = 0$, each of the two users measures their respective qubits of the i -th shared state using the POVM $P^0 = \{P_0^0, P_1^0, P_2^0\}$ where

$$\begin{aligned}
 P_0^0 &\equiv \frac{(\sin\theta|0\rangle - \cos\theta|1\rangle)(\sin\theta\langle 0| - \cos\theta\langle 1|)}{1 + \cos\theta} \\
 P_1^0 &\equiv \frac{1}{1 + \cos\theta}|1\rangle\langle 1| \\
 P_2^0 &\equiv I - M_0^0 - M_1^0
 \end{aligned}$$

- For $a_i = 1$, each of the two users measures their respective qubits of the i -th shared state using the POVM $P^1 = \{P_0^1, P_1^1, P_2^1\}$ where

$$\begin{aligned}
 P_0^1 &\equiv \frac{(\cos\theta|0\rangle + \sin\theta|1\rangle)(\cos\theta\langle 0| + \sin\theta\langle 1|)}{1 + \cos\theta} \\
 P_1^1 &\equiv \frac{1}{1 + \cos\theta}|0\rangle\langle 0| \\
 P_2^1 &\equiv I - M_0^1 - M_1^1
 \end{aligned}$$

- If an user gets $P_0^0(P_1^0)$ for $a_i = 0$, he guesses the original i -th raw key bit as 0(1). Whenever he gets P_2^0 , his guess remains inconclusive.
- Similarly, if an user obtains $P_0^1(P_1^1)$ for $a_i = 1$, he guesses the original i -th raw key bit as 0(1). If he gets P_2^1 , his guess remains inconclusive.

- Suppose for $N = 3$, after the key generation phase, the raw key bits generated at the server's side is $0\ 1\ 1\ 0\ 1\ 1$ and at any one of the user's side is $?\ ?\ 1\ 0\ ?\ ?$ (here $?$ denotes unknown bits).
- If the server and the user XOR every two consecutive raw key bits to generate a final key of length 3 bits, then the final key at the server's side will be $1\ 1\ 0$ and the user's side will be $?\ 1\ ?$.
- If the user wants to retrieve (say) the 3rd database bit, she announces a 1-bit right shift to the server. The server then encrypts the database with the shifted key and sends the encrypted database to the user, from which the user can retrieve the intended bit.

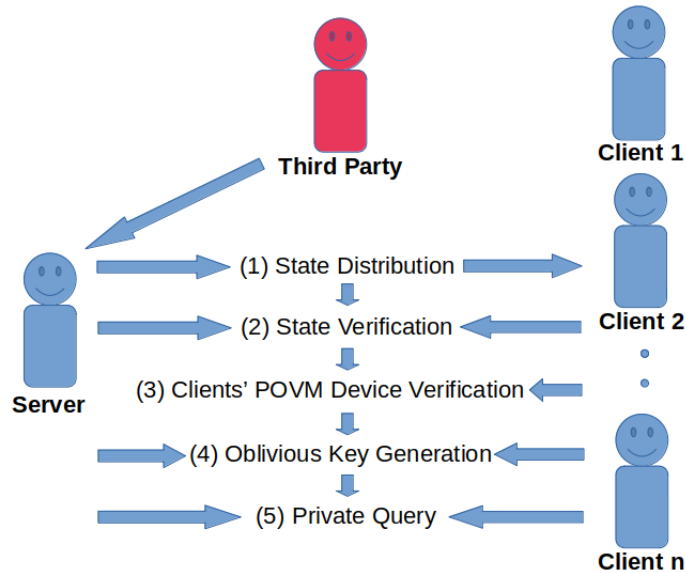


Figure 7-1: Schematic diagram of different phases of our multi-client Semi-DI QPQ scheme

7.4 Analysis of our proposal

Here, we discuss about the performance of our proposal. First, we look into the correctness issue of this scheme, and then we dive into the security aspects of this proposal.

7.4.1 Correctness of the scheme

Here, we demonstrate the accuracy of the protocol.

Theorem 15. *If this multi-user QPQ proposal is implemented honestly, then after the shared key generation phase, every client can accurately retrieve (on average) around $(1 - \cos \theta)$ fraction of bits from the actual raw key R (held by the server).*

Proof. After the *shared key generation phase*, the server and the clients share a kN -bit raw key such that the server knows all the bits and each of the clients knows only some of the bits. According to the protocol, these raw key bits generation can be redefined as follows.

The server first generates a kN bit random string $R = r_1 \dots r_{kN}$ (which is the raw key at the server's side, i.e., $R_j = r_j$). For $r_j = 0$, the server performs an identity operator \mathbb{I}_2 to each qubits of the j -th $(n+1)$ -qubit GHZ state. Otherwise (i.e., for $r_j = 1$), the server performs an unitary operator U to each qubits of the j -th $(n+1)$ -qubit GHZ state which transforms $|0\rangle$ to $|0'\rangle$ and $|1\rangle$ to $|1'\rangle$ where $|0'\rangle = (\cos \theta|0\rangle + \sin \theta|1\rangle)$ and $|1'\rangle = (\sin \theta|0\rangle - \cos \theta|1\rangle)$.

After these operations, the server shares the states with the n -clients such that the j -th client receives the j -th qubit of the shared state, and the server holds the $(n+1)$ -th qubit of each shared state. After sharing the states, the server measures all his qubits having $r_j = 0$ in $\{|0\rangle, |1\rangle\}$ basis and the rest of the qubits (for which $r_j = 1$) in $\{|0'\rangle, |1'\rangle\}$ basis. After measuring his share of the j -th state, if he gets $|0\rangle$ or $|0'\rangle$ at his side, he announces $a_j = 0$, otherwise (if he gets $|1\rangle$ or $|1'\rangle$), he announces $a_j = 1$. Now, the client's job is to guess the value of each r_j .

Whenever the server declares $a_j = 0$, all the clients can understand that the server gets either $|0\rangle$ or $|0'\rangle$ and their qubits for the j -th shared state also collapses to $|0\rangle$ or $|0'\rangle$ respectively. However, to obtain the exact value of the corresponding raw key bit, each client must distinguish these two states with certainty. As $|0\rangle$ and $|0'\rangle$ are non-orthogonal states (when $\theta \neq \frac{\pi}{2}$), the clients cannot distinguish these two states with certainty for all the instances.

According to the procedure described in our proposal, whenever the server declares $a_j = 0$, each client chooses the POVM $\{P_0^0, P_1^0, P_2^0\}$. After measurement, if a client gets the outcome P_0^0 (P_1^0), she concludes that the outcome at the server's side is $|0\rangle$ ($|0'\rangle$) for the j -th shared instance and the corresponding raw key bit at the server's side is $r_j = 0$ ($r_j = 1$). However, if a client receives the outcome P_2^0 , she remains uncertain about the value of the raw key bit corresponding to the j -th instance at the server's side. In this similar way, the clients can also conclude about the raw key bits at the server's side for $a_j = 1$.

Now, we calculate the success probabilities of getting different outcomes for the instances having $a_j = 0$ (i.e., for the input states $|0\rangle$ and $|0'\rangle$).

For input state $|0\rangle$, the success probabilities of getting different outcomes will be,

$$\begin{aligned}
\Pr(P_0^0||0\rangle) &= \langle 0|P_0^0|0\rangle \\
&= (1 - \cos \theta) \\
\Pr(P_1^0||0\rangle) &= \langle 0|P_1^0|0\rangle \\
&= 0 \\
\Pr(P_2^0||0\rangle) &= \langle 0|P_2^0|0\rangle \\
&= \cos \theta
\end{aligned}$$

Similarly, for the state $|0'\rangle$, the success probabilities will be

$$\begin{aligned}
\Pr(P_0^0||0'\rangle) &= \langle 0'|P_0^0|0'\rangle \\
&= 0 \\
\Pr(P_1^0||0'\rangle) &= \langle 0'|P_1^0|0'\rangle \\
&= (1 - \cos \theta) \\
\Pr(P_2^0||0'\rangle) &= \langle 0'|P_2^0|0'\rangle \\
&= \cos \theta
\end{aligned}$$

Whenever the server declares $a_j = 1$, every client chooses the POVM $\{P_0^1, P_1^1, P_2^1\}$. For these instances, in a similar way as discussed earlier, we can calculate the corresponding success probabilities of getting different outcomes. We formalize all the conditional probabilities in the following table.

a_j	Conditional Probability of any i -th Client			
	Client Server	$C_i = P_0^0/P_0^1$	$C_i = P_1^0/P_1^1$	$C_i = P_2^0/P_2^1$
0	$S = 0\rangle$	$1 - \cos \theta$	0	$\cos \theta$
0	$S = 0'\rangle$	0	$1 - \cos \theta$	$\cos \theta$
1	$S = 1\rangle$	$1 - \cos \theta$	0	$\cos \theta$
1	$S = 1'\rangle$	0	$1 - \cos \theta$	$\cos \theta$

According to the protocol, if $a_j = 0$ and the i -th client gets $P_0^0(P_1^0)$, she considers $R_{C_j}^i = 0(1)$. When $a_j = 1$ and the i -th client gets $P_0^1(P_1^1)$, she considers $R_{C_j}^i = 0(1)$. Thus, the success probability of any i -th client to guess the j -th raw key bit R_j (where $R_j = r_j$) of the server can be written as

$$\begin{aligned}
&\Pr(R_{C_j}^i = R_j) \\
&= \Pr(R_{C_j}^i = 0, R_j = 0) + \Pr(R_{C_j}^i = 1, R_j = 1) \\
&= (1 - \cos \theta).
\end{aligned}$$

That means in this proposal, the overall success probability for each of the clients in guessing a raw key bit is equal to $(1 - \cos \theta)$. So, at the end of the *shared key generation phase*, each of the clients can successfully retrieve (on average) around $(1 - \cos \theta)$ fraction of bits from the actual raw key with certainty. \square

7.4.2 Estimation of parameters for private query phase

In this subsection, we will derive the parameter values for an honest implementation of this multi-client proposal while ensuring that both the privacy of the clients and the privacy of the server are preserved.

Estimation of the security parameter θ :

Like the recent full DI-QPQ proposal in [15], here also, the server wants each of the clients to know at least one and always less than two final key bits for database security. The result in Theorem 15 shows that under the proposed method, each client has a probability of approximately $(1 - \cos \theta)$ of correctly guessing a raw key bit. In generating the final key, each client XORs k raw key bits. So, the probability of every client correctly guessing a final key bit is approximately $(1 - \cos \theta)^k$.

If we consider that f_{c_i} denotes the i -th client's known final key bits then, then the expected value of f_{c_i} will be,

$$E[f_{c_i}] \approx (1 - \cos \theta)^k N. \quad (7.1)$$

For security purposes, the server wants each of the clients to know between one to two final key bits. That means for any i -th client, the following condition must be satisfied.

$$1 \leq E[f_{c_i}] < 2.$$

This implies that,

$$\begin{aligned} 1 &\leq (1 - \cos \theta)^k N < 2 \\ \frac{1}{N} &\leq (1 - \cos \theta)^k < \frac{2}{N}. \end{aligned} \quad (7.2)$$

Based on the results presented above, the following conclusion can be drawn.

Corollary 14. *To ensure that every client knows at least one but less than two final key bits, the server must select the value of θ such that,*

$$\frac{1}{N} \leq (1 - \cos \theta)^k < \frac{2}{N}.$$

Estimation of P_a and P_c for security purpose:

From the correctness result in theorem 15, we can conclude that the probability of each client not guessing any final key bit correctly will be,

$$\begin{aligned} \Pr(\text{no final key bit for a client}) &\approx [1 - (1 - \cos \theta)^k]^N \\ &\approx e^{-(1 - \cos \theta)^k N}. \end{aligned} \quad (7.3)$$

As we assume a total of n number of clients in our multi-client proposal, the probability that none of the clients know any final key bits is equal to,

$$\Pr(\text{no final key bit for all } n \text{ clients}) \approx e^{-(1 - \cos \theta)^k n N}. \quad (7.4)$$

For this multi-client proposal, we assume that the scheme aborts whenever none of the clients know any of the final key bits. So, from the definition 2, we can conclude that the parameter P_a will be upper bounded by,

$$P_a \leq e^{-(1 - \cos \theta)^k n N}. \quad (7.5)$$

If we substitute $(1 - \cos \theta)^k$ in this relation from equation 7.2, assuming that the server chooses θ such that $(1 - \cos \theta)^k = \frac{1}{N}$, then from equation 7.5, we get,

$$\boxed{P_a \leq e^{-n}}. \quad (7.6)$$

This suggests that the value of P_a is small for this multi-client proposal (as $n > 1$). So, the probability that this multi-client scheme does not terminate in the honest scenario is equal to,

$$\begin{aligned} \Pr(\text{scheme doesn't terminate in honest scenario}) \\ \geq (1 - e^{-n}). \end{aligned} \quad (7.7)$$

Hence, for this multi-client proposal, the likelihood of the scheme not terminating is high.

Now, to derive a bound on the security parameter P_c , we first refer to the Chernoff-Hoeffding inequality [59], which is already mentioned in Chapter 4 Proposition 1.

For any i -th client in our proposal, we define $X_j^i = 1$ if the j -th final key bit is known to the client (i.e., she gets conclusive POVM outcome for the j -th instance), and $X_j^i = 0$ otherwise. We can define the random variable X^i for any i -th client as the sum of the X_j^i values, where j ranges from 1 to the total number of final key bits, which is N . From the correctness result in theorem 15, we can say that in the honest scenario, the expected number of final key bits that any i -th client should know is $Y^i = (1 - \cos \theta)^k N$.

For this proposal, the collapsed states at each of the client's sides are independent, and we assume that the clients also apply their measurement devices independently for each qubit. So, to ensure the value X^i to lie within the deviation of $\delta_t = \epsilon (1 - \cos \theta)^k N$ for any i -th client from the expected value, here we can use the

Chernoff-Hoeffding inequality. The value of each X^i and the corresponding Y^i are calculated here under the assumption that the scheme does not terminate. So, using the expression for the Chernoff-Hoeffding bound from Proposition 1, it is clear that for every i -th client,

$$\begin{aligned} & \Pr [|X^i - Y^i| < \delta_t \wedge \text{scheme doesn't terminate}] \\ & \geq 1 - \exp(-2\delta_t^2 m). \end{aligned} \quad (7.8)$$

The *shared key generation phase* results in N final key bits shared by the server and each of the clients. Among those N bits, the server aims the total final key bits known to every i -th client fall within $[p - \epsilon p, p + \epsilon p]$, where $p = (1 - \cos \theta)^k N$ and the allowed deviation is $\delta_t = \epsilon (1 - \cos \theta)^k N$. Using this relation in equation 7.8 with δ_t and m values, one can get,

$$\begin{aligned} & \Pr [|X^i - Y^i| < \delta_t \wedge \text{scheme doesn't terminate}] \\ & \geq 1 - \exp(-2\delta_t^2 N) \\ & \text{where } \delta_t = \epsilon (1 - \cos \theta)^k N. \end{aligned} \quad (7.9)$$

From equation 7.2, it is clear that for this multi-client proposal, the server selects the parameters θ and k such that $\frac{1}{N} \leq (1 - \cos \theta)^k < \frac{2}{N}$. If the server chooses θ and k for which $(1 - \cos \theta)^k = \frac{1}{N}$, then substituting this relation into equation 7.9 will yield,

$$\begin{aligned} & \Pr [|X^i - Y^i| < \epsilon \wedge \text{scheme doesn't terminate}] \\ & \geq 1 - \exp(-2\epsilon^2 N). \end{aligned} \quad (7.10)$$

In our proposed method, to encrypt the database, the server utilizes a bitwise XOR operation with the final key. Therefore, if a client correctly guesses a final key bit, it implies the correct guess of the corresponding database bit. So, as per definition 1, it can be concluded that when the server and every i -th client are honest, the lower bound of the parameter P_c in our proposal is determined by,

$$P_c \geq [1 - \exp(-2\epsilon^2 N)]. \quad (7.11)$$

That means the likelihood of every i -th client knowing the expected number of data bits while the scheme not terminating is high in the honest scenario of our proposal, as N is large in practice.

The server chooses θ and k so that every client knows between one to two final key bits. So, the deviation, δ_t , has the following bound according to equation 7.2.

$$\epsilon \leq \delta_t < 2\epsilon. \quad (7.12)$$

That means the upper bound on ϵ can be derived from $2\epsilon \leq 1$, yielding $\epsilon \leq \frac{1}{2}$.

We now address the security concerns regarding this proposed multi-user QPQ scheme.

7.4.3 Security issues of the scheme

The security issues of this proposal can be divided into three phases, namely security in (semi)-device independent scenario, security of the database against dishonest clients, and security of the clients against dishonest servers. The security of this proposal mainly follows from the results mentioned in [82] and [15]. Here, we point out those exact results and discuss how they guarantee the security of this proposal.

Security in (semi)-device independent scenario

In this scheme, the device certification has been done in two phases, namely *entangled state verification phase* and *client's POVM device verification phase*.

In the $\text{GHZtest}()$ of *entangled state verification phase*, each of the participants acts as a verifier in subsequent iterations to certify the states provided by the third party. The participant who acts as a verifier in a particular iteration first selects some of the states randomly from their shared instances. After that, the server declares his chosen r_j values for each of those selected instances so that each of the participants can apply the corresponding unitaries in their respective qubits (that were applied by the server before the *entangled state supply phase*) to get back the actual states supplied by the third party. The verifier then chooses a random $(n + 1)$ -bit input string for each of his chosen states, declares those input bits for the other participants so that each participant can measure their corresponding qubits on a specified basis, and sends the corresponding outcomes to the verifier. The verifier then checks whether their measurement angles and the corresponding outputs satisfy a predefined relation. In this phase, all the participants (i.e., the server and the clients) want to certify the states provided by the third party. So, all of them must act honestly (according to assumption 5) in this phase.

The results mentioned in [82] and [115] lead us to the following conclusion.

Corollary 15 (Verification of the input states). *The GHZtest in entangled state verification phase either certifies that the inputs and the outputs corresponding to the chosen states satisfy the relation mentioned in this test, meaning the given states are $(n + 1)$ -qubit GHZ states (or the unitary equivalent of them), or the scheme is likely to abort in the long run.*

In the next device certification phase (i.e., in *client's POVM device verification phase*), the clients check the functionality of their measurement devices. After the *entangled state supply phase*, each of the client's goals is to distinguish between two non-orthogonal states (either between $|0\rangle$ and $|0'\rangle$ or between $|1\rangle$ and $|1'\rangle$) with as much probability as possible. For this reason, each client chooses between one of the two unknown devices (depending on the mentioned a_i values) and verifies whether these devices attain the expected values of Ω^0 and Ω^1 . One may check that this distinction procedure (at the client's side) is the same as the procedure followed by the client in [15]. For this reason, from the results mentioned in [15, theorem 3 and theorem 4], we can conclude the following for this proposal.

Corollary 16 (Verification of the client’s devices). *The ClientPOVMtest in client’s POVM device verification phase either certifies that each of the client’s measurement devices attains the intended values of Ω^0 and Ω^1 , i.e., the devices at each of the client’s side are of the specified form (up to a local unitary), or the scheme is likely to abort in the long run.*

The results mentioned in corollary 15 and corollary 16 boil down to the following conclusion.

Corollary 17. *Our semi-DI proposal either terminates with a high likelihood in the asymptotic limit or confirms that the devices in the GHZtest and the ClientPOVMtest achieve the intended values of the parameters in the respective testing phases.*

Note that our proposal doesn’t certify the devices that perform certain unitary operations (one device performs identity operation, and the other device performs θ rotation over the qubits) at the server’s and each of the client’s side. Additionally, the scheme also doesn’t certify the projective measurement devices on the server’s side. For this reason, our proposed scheme is semi-device independent. However, to ensure full device independence, one may use a process tomography technique to certify the unknown unitary devices at the server’s and clients’ side and can adopt the measurement device certification technique mentioned in [15] that follows the approach of [65] for certifying non-maximally incompatible observables.

Security of the database against dishonest clients

In this subsection, we mention the number of raw key bits that each of the dishonest clients can guess in the *shared key generation phase* of this proposal. As the raw key bit generation procedure at the client’s side is the same here as the client’s raw key generation procedure in the QPQ scheme [15], the results related to the cheating of the dishonest client in [15] will exactly follow here.

Corollary 18. *In the absence of ClientPOVMtest in our proposal, each of the clients can inconclusively retrieve atmost $(\frac{1}{2} + \frac{1}{2} \sin \theta)$ fraction of bits from the server’s raw key during the shared key generation phase.*

Similar to the proposal in [15], here also we introduce a testing phase for the clients’ measurement devices before the *shared key generation phase*. So, from the result in [15, Lemma 1], here we can conclude the following.

Corollary 19. *For this QPQ proposal, either the scheme terminates with a high likelihood (as the limit approaches infinity) or each of the dishonest clients can retrieve (on average) at most $(1 - \cos \theta)$ fraction of bits from the entire raw key (at the server’s side) in the shared key generation phase.*

In the *private query phase*, k raw key bits are XOR-ed to construct every bit of the final key. As the guessing probability of a dishonest client about a raw key bit is at most $(1 - \cos \theta)$ (according to the result in Corollary 19) and the dishonest clients also process each of the raw key bits independently (as mentioned in assumption 4),

the maximum guessing probability of a dishonest client for a final key bit will be $(1 - \cos \theta)^k$. So, from the definition 3, we can say that the fraction τ of the data bits that each of the clients can guess correctly in a single query to the database will be,

$$\tau \leq (1 - \cos \theta)^k. \quad (7.13)$$

Now, substituting $(1 - \cos \theta)^k$ using the upper limit achieved in the equation 7.2, the following bound can be obtained on τ .

$$\boxed{\tau < \frac{2}{N}}. \quad (7.14)$$

It implies that τ is significantly smaller than N for this multi-client proposal.

Security of the clients against dishonest server

In this subsection, we estimate the number of indices that the dishonest server can guess from a particular client's query indices (this result will follow for all the clients). Here also, like the QPQ proposal [15], none of the clients have declared anything regarding their measurement outcomes and, consequently, their known raw key (or final key) bits. So, similar to the result in [15, Lemma 2], here we can conclude the following.

Corollary 20. *If a client retrieves l many data bits from the N -bit database using l_q many queries, then the server can predict whether a particular data bit is retrieved by the client with likelihood around $\frac{l}{N}$.*

This result implies that if the server desires to obtain a client's query indices with greater certainty, the server must permit the client to retrieve more than l data bits in l_q queries, which contradicts our assumption (specifically, assumption 5).

The result in corollary 20 implies that in l many guesses, the expected number of indices that dishonest server (\mathcal{S}^*) can predict correctly from a client's query index set is,

$$\begin{aligned} E[\mathcal{I}_{\mathcal{S}^*}] &= \Pr(\text{Server predicts an index correctly in a single guess}) \cdot l \\ &\approx \frac{l^2}{N}. \end{aligned} \quad (7.15)$$

From the relation in equation 7.15 and from the definition 4, we can argue that for every client, the value of δ will be,

$$\delta \leq \left(\frac{l}{N} \right). \quad (7.16)$$

In practice, the database size, N , is significantly larger than the size l of a client's query index set, with N approximately equal to l^n for some positive integer n . From

this information and the relation in equation 7.16, the upper limit on δ can be written as,

$$\delta \leq \frac{1}{l^{(n-1)}}. \quad (7.17)$$

That means the value of δ is significantly smaller than l in our multi-client proposal.

7.5 Discussion and Conclusion

Most existing QPQ proposals are limited to the single-user scenario, which is inefficient when multiple users are involved. Recent multi-user proposals rely on a semi-trusted server and only consider the retrieval of items of common interest, making them impractical. To overcome these limitations, in this chapter, we propose a semi-device independent multi-user QPQ scheme where each user can retrieve different items simultaneously without revealing their choices to others or relying on a semi-trusted server. Our scheme allows each user to retrieve optimal raw key bits during the oblivious key generation phase. Unlike existing proposals that only consider certain eavesdropping strategies, we formally evaluate the security issues and derive upper limits on the likelihood of cheating for both the server and users.

“I may not have gone where I intended to go, but I think I have ended up where I needed to be..”

— Douglas Adams, *The Long Dark Tea-Time of the Soul*.

In this concluding chapter, we take a comprehensive look at the previous chapters, summarizing and drawing conclusions from the various aspects explored in this thesis. The primary focus of this research has been on the QPQ primitive in the DI scenario. Here, we highlight our key contributions, improvements, and extensions to existing methods. Furthermore, we delve into the potential directions for future research and identify the open problems that lie ahead in the field of QPQ.

8.1 Summary of technical results

Chapter 1 served as the thesis introduction, while Chapter 2 provided a foundational understanding of quantum information and computation, laying the groundwork for readers to read the thesis comfortably. In Chapter 3, an overview of the QPQ primitive was presented, encompassing its evolution and its relationship with other related primitives. This chapter also delved into the crucial aspects of security definitions and the necessary assumptions for analyzing the proposals related to this thesis. The primary technical outcomes of the thesis were then explored in Chapters 4, 5, 6 and 7, and the highlights of these chapters are as follows.

Like most initial quantum cryptography schemes, the security of the initial QPQ schemes also relies on the functionality of the involved devices. Later, it was shown that if those devices do not work accordingly, some information may leak to the adversary. Maitra et al. [77] first introduced DI in the QPQ domain by proposing a semi-DI version of the QPQ scheme [117] to address these assumptions. In Chapter 4, we move one step further and propose a novel fully DI-certified QPQ scheme using maximally entangled states for improved robustness. Our scheme achieves the optimal number of raw key bits (exploiting the optimal POVM measurement for distinguishing two

non-orthogonal states) for the client Alice in the oblivious key generation phase. We analyze security issues formally against all attacks, preserving the correctness condition. We provide upper bounds on the cheating probabilities for both the dishonest client and server. This new QPQ scheme, incorporating QKD, can potentially become a crucial near-term application of the quantum internet.

Maitra et al. [77] first identified that the security of the existing QPQ schemes (up until that time) relies on the functionality of the devices. As an example, they considered the QPQ scheme [117] and suggested a tilted version of the actual CHSH test locally (at the server side) on top of the QPQ scheme [117] to certify the functionality of the devices. However, their proposal is semi-DI as the local test on the server side does not certify the functionality of the client’s measurement device. This issue is already discussed in Chapter 4. In Chapter 5, we exploit the proper self-testing mechanism of observables along with the local version of the tilted CHSH test to certify the functionality of all the devices involved in the QPQ scheme [117]. We compare the performance of this full DI proposal of the QPQ scheme [117] with the performance of our full DI-QPQ proposal in Chapter 4 and *discuss relative advantages of both these schemes*. Inspired by the proposal in Chapter 4, in this chapter, we further propose a DI scheme for a modification of [117] where the client can retrieve the maximum conclusive raw key bits. In summary, based on the assumptions discussed in Chapter 3, in this chapter, we have strengthened the security of the QPQ scheme [117] and also enhanced the performance (in the modified proposal).

Chapter 6 deals with several strategies to reduce the overall sample size required (in finite sample scenario) for the DI testing phase in [77]. In this chapter, we derive the relation between the required sample size and corresponding success probability and propose optimal testing mechanisms for DI-QPQ proposal [77]. CHSH tests on different versions of the entangled states are studied. Further, we consider the three-party Pseudo Telepathy as a tool for testing DI and show that it provides significantly better results for practical purposes.

Chapter 7 deals with the scenario where multiple users are involved in a QPQ scheme. Most of the existing QPQ proposals are limited to the single-user scenario, which is inefficient when multiple users are involved. Recent multi-user proposals rely on a semi-trusted server and only consider the retrieval of items of common interest, making them impractical. To overcome these limitations, in this chapter, we propose a semi-device independent multi-user QPQ scheme where each user can retrieve different items simultaneously without revealing their choices to others or relying on a semi-trusted server. Our scheme allows each user to retrieve optimal raw key bits during the oblivious key generation phase. Unlike the existing multi-user proposals that only consider certain eavesdropping strategies, we formally evaluate the security issues and derive upper limits on the likelihood of cheating for both the server and users.

8.2 Possible future works

Here, we list some open problems and possible future works in the domain of QPQ.

- **Reduction of classical communication complexity** : The main limitations in all the existing QPQ proposals are that they have a huge classical communication complexity as the server requires sending the entire encrypted database to each client for every single query. Reducing this classical communication complexity will be an interesting future work in this direction.
- **Consideration of non-i.i.d. scenario** : All the schemes mentioned in this thesis consider the assumption that the devices involved in the proposals follow the *i.i.d* assumption, which is not very practical. Recently, there are some results for multi-round protocols on bit commitment [12], oblivious transfer and bit commitment [43], weak string erasure [66] etc. without the *i.i.d.* assumption. Although in [43] and [66], there are bounded/noisy storage assumptions. There are also some results in the single-shot setting (where the *i.i.d.* assumption is irrelevant) on bit commitment and coin flipping [104], weak coin flipping [11], XOR oblivious transfer [73] etc. However, to the best of our knowledge, there is still no result on the DI scenario of the distrustful primitive QPQ without the *i.i.d* assumption. So, analysis of the full DI schemes proposed in this thesis without considering the *i.i.d.* assumption will be an interesting research problem in the domain of QPQ.
- **Consideration of noise parameter** : The schemes proposed in this thesis consider the asymptotic scenario where no channel noise exists. However, in practice, these protocols will be executed in a noisy environment considering some finite number of samples. So, for the practical scenario, the analysis of these schemes considering channel noise and a finite number of samples can be a possible future work in this direction.

Bibliography

- [1] Commitment Scheme. <https://www.cs.princeton.edu/courses/archive/spr08/cos598D/scribe5.pdf>, .
- [2] Commitment Scheme. <http://theory.stanford.edu/~trevisan/cs276/lecture27.pdf>, .
- [3] Enigma Machine. https://en.wikipedia.org/wiki/Enigma_machine, .
- [4] (Symmetric) Private Information Retrieval. [https://wiki.veriqloud.fr/index.php?title=\(Symmetric\)_Private_Information_Retrieval](https://wiki.veriqloud.fr/index.php?title=(Symmetric)_Private_Information_Retrieval), .
- [5] Vernam Cipher. https://en.wikipedia.org/wiki/Gilbert_Vernam, .
- [6] Vigenere Cipher. https://en.wikipedia.org/wiki/Vigenere_cipher, .
- [7] A. Acín, N. Brunner, N. Gisin, S. Massar, S. Pironio, and V. Scarani. Device-independent security of quantum cryptography against collective attacks. *Phys. Rev. Lett.*, 98(23):230501, 2007.
- [8] A. Acín, S. Massar, and S. Pironio. Randomness versus Nonlocality and Entanglement. *Phys. Rev. Lett.*, 108:100402, 2012.
- [9] E. Adlam and A. Kent. Device-independent relativistic quantum bit commitment. *Phys. Rev. A*, 92:022315, 2015.
- [10] L. M. Adleman. On breaking the iterated Merkle-Hellman public-key cryptosystem. In *Advances in Cryptology CRYPTO 1983*, pages 303–308, 1983.
- [11] N. Aharon, A. Chailloux, I. Kerenidis, S. Massar, S. Pironio, and J. Silman. Weak coin flipping in a device-independent setting. In *Theory of Quantum Computation, Communication, and Cryptography. TQC 2011.*, pages 1–12, 2011.
- [12] N. Aharon, S. Massar, S. Pironio, and J. Silman. Device-independent bit commitment based on the CHSH inequality. *New J. of Phys.*, 18 (2):025014, 2016.

- [13] D. Aharonov, Z. Brakerski, K-M Chung, A. Green, C-Y Lai, and O. Sattath. On quantum advantage in information theoretic single-server PIR. In *Advances in Cryptology – EUROCRYPT 2019*, page 219–246, 2019.
- [14] C. Bamps and S. Pironio. Sum-of-squares decompositions for a family of Clauser-Horne-Shimony-Holt-like inequalities and their application to self-testing. *Phys. Rev. A*, 91:052111, 2015.
- [15] J. Basak, K. Chakraborty, A. Maitra, and S. Maitra. A Proposal for Device Independent Probabilistic Quantum Oblivious Transfer. In *LNCS 13774, INDOCRYPT 2022, Springer*, pages 541–565, 2022 (Full version available at <https://arxiv.org/abs/1901.03042>).
- [16] J. Basak and S. Maitra. Clauser-Horne-Shimony-Holt versus three-party pseudo-telepathy: on the optimal number of samples in device-independent quantum private query. *Quant. Inf. Process.*, 17:77, 2018.
- [17] Á. Baumeler and A. Broadbent. Quantum private information retrieval has linear communication complexity. *J. Cryptol.*, 28:161–175, 2015.
- [18] M. Ben-Or, S. Goldwasser, J. Kilian, and A. Wigderson. Multi-prover interactive proofs: How to remove intractability assumptions. In *In Proceedings of the twentieth annual ACM symposium on Theory of computing, STOC 1988*, pages 113–131, 1988.
- [19] C. H. Bennett. Quantum cryptography using any two nonorthogonal states. *Phys. Rev. Lett.*, 68 (21):3121–3124, 1992.
- [20] C. H. Bennett, F. Bessette, G. Brassard, L. Salvail, and J. Smolin. Experimental quantum cryptography. *J. Cryptology*, 5(1):3–28, 1992.
- [21] C. H. Bennett and G. Brassard. Quantum cryptography: Public key distribution and coin tossing. In *In Proceedings of IEEE International Conference on Computers, Systems and Signal Processing*, pages 175–179, 1984.
- [22] C. H. Bennett, G. Brassard, S. Breidbart, and S. Wiesner. Quantum cryptography, or unforgeable subway tokens. In *In Advances in Cryptology-CRYPTO 1983*, page 267–275, 1983.
- [23] C. H. Bennett, G. Brassard, C. Crépeau, and M. H. Skubiszewska. Practical quantum oblivious transfer. In *In Advances in Cryptology-CRYPTO 1991*, page 576, 1991.
- [24] C. H. Bennett, G. Brassard, and N. D. Mermin. Quantum Cryptography without Bell’s Theorem. *Phys. Rev. Lett.*, 68(5):557–559, 1992.
- [25] J. A. Bergou, E. Feldman, and M. Hillery. Optimal unambiguous discrimination of two subspaces as a case in mixed-state discrimination. *Physical Review A*, 73(3):032107, 2006.

- [26] M. Blum. Coin Flipping by Telephone. In *Advances in Cryptology: A Report on CRYPTO 1981*, pages 11–15, 1981.
- [27] J. Bouda, M. Pawłowski, M. Pivoluska, and M. Plesch. Device-independent randomness extraction for arbitrarily weak min-entropy source. *Phys. Rev. A*, 90:032313, 2014.
- [28] G. Brassard, A. Broadbent, and A. Tapp. Quantum pseudo-telepathy. *Found. Phys.*, 35 (11):1877–1907, 2005.
- [29] G. Brassard, D. Chaum, and C. Crépeau. Minimum Disclosure Proofs of Knowledge. *J. Comput. System Sci.*, 37(2):156–189, 1988.
- [30] G. Brassard and C. Crépeau. Quantum bit commitment and coin tossing protocols. In *In Advances in Cryptology-CRYPTO 1990*, page 537, 1990.
- [31] A. Broadbent and P. Yuen. Device-independent oblivious transfer from the bounded-quantum-storage-model and computational assumptions. *New J. Phys.*, 25:053019, 2023.
- [32] D. R. L. Brown. Breaking RSA May Be As Difficult As Factoring. *Journal of Cryptology*, 29:220–241, 2016.
- [33] H. Buhrman, N. Chandran, S. Fehr, R. Gelles, V. Goyal, R. Ostrovsky, and C. Schaffner. Position-based quantum cryptography: Impossibility and constructions. In *In Advances in Cryptology-CRYPTO 2011*, page 429–446, 2011.
- [34] H. Buhrman, M. Christandl, and C. Schaffner. Complete insecurity of quantum protocols for classical two-party computation. *Phys. Rev. Lett.*, 109(16):160501, 2012.
- [35] K. Chakraborty, A. Chailloux, and A. Leverrier. Arbitrarily Long Relativistic Bit Commitment. *Phys. Rev. Lett.*, 115:250501, 2015.
- [36] N. Chandran, V. Goyal, R. Moriarty, and R. Ostrovsky. Position based cryptography. In *In Advances in Cryptology-CRYPTO 2009*, pages 391–407, 2009.
- [37] A. Chefles and S. M. Barnett. Optimum unambiguous discrimination between linearly independent symmetric states. *Physics Letters A*, 250(4):223–229, 1998.
- [38] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan. Private information retrieval. In *In 36th Annual Symposium on Foundations of Computer Science FOCS 1995*, pages 41–50, 1995.
- [39] B. S. Cirel’son. Quantum generalizations of Bell’s inequality. *Lett. in Math. Phys.*, 4(2):93–100, 1980.
- [40] J. F. Clauser, M. A. Horne, A. Shimony, and R. A. Holt. Proposed experiment to test local hidden-variable theories. *Phys. Rev. Lett.*, 23:880–884, 1969.

- [41] R. Colbeck. Impossibility of secure two-party classical computation. *Phys. Rev. A*, 76(6):062308, 2007.
- [42] R. Colbeck and A. Kent. Private randomness expansion with untrusted devices. *J. Phys. A: Math. Theor.*, 44:095305, 2010.
- [43] I. B. Damgaard, S. Fehr, R. Renner, L. Salvail, and C. Schaffner. A tight high-order entropic quantum uncertainty relation with applications. In *LNCS 4622, CRYPTO 2007*, page 360–378, 2007.
- [44] I. B. Damgard, S. Fehr, L. Salvail, and C. Schaffner. Cryptography in the bounded quantum-storage model. In *In IEEE Information Theory Workshop on Theory and Practice in Information-Theoretic Security*, page 24–27, 2005.
- [45] W. Diffie and M. Hellman. New directions in cryptography. *IEEE Trans. Inf. Theor.*, 22(6):644–654, 2006.
- [46] A. K. Ekert. Quantum Cryptography Based on Bell’s Theorem. *Phys. Rev. Lett.*, 67(6):661–663, 1991.
- [47] A. K. Ekert and R. Renner. The ultimate physical limits of privacy. *Nature*, 507(7493):443–447, 2014.
- [48] S. Fehr and M. Fillinger. On the Composition of Two-Prover Commitments, and Applications to Multi-round Relativistic Commitments. In *Advances in Cryptology – EUROCRYPT 2016*, pages 477–496, 2016.
- [49] C. A. Fuchs and J. V. de Graaf. Cryptographic distinguishability measures for quantum-mechanical states. *IEEE Trans. Inf. Theory*, 45:1216, 1999.
- [50] F. L. Gall. Quantum private information retrieval with sublinear communication complexity. *Theory of Computing*, 8 (16):369–374, 2012.
- [51] F. Gao, B. Liu, W. Huang, and Q. Y. Wen. QKD-based quantum private query without a failure probability. *Sci. China-Phys. Mech. Astron.*, 58:100301, 2015.
- [52] F. Gao, B. Liu, Q. Y. Wen, and H. Chen. Flexible quantum private queries based on quantum key distribution. *Opt. Express*, 20:17411–17420, 2012.
- [53] Y. Gertner, Y. Ishai, E. Kushilevitz, and T. Malkin. Protecting data privacy in private information retrieval schemes. In *In the thirtieth annual ACM symposium on Theory of computing STOC 1998*, pages 151–160, 1998.
- [54] V. Giovannetti, S. Lloyd, and L. Maccone. Quantum Private Queries. *Phys. Rev. Lett.*, 100(23):230502, 2008.
- [55] V. Giovannetti, S. Lloyd, and L. Maccone. Quantum Private Queries: Security Analysis. *IEEE Trans. Info. Theory*, 56 (7):3465–3477, 2010.

- [56] N. Gisin, S. Fasel, B. Kraus, H. Zbinden, and G. Ribordy. Trojan-horse attacks on quantum-key-distribution systems. *Phys. Rev. A*, 73(2):022320, 2006.
- [57] N. Gisin, S. Pironio, and N. Sangouard. Proposal for implementing device-independent quantum key distribution based on a heralded qubit amplifier. *Phys. Rev. Lett.*, 105:070501, 2010.
- [58] C. W. Helstrom. *Quantum detection and estimation theory (Mathematics in science and engineering series)*. 123, Academic Press, New York, 1976.
- [59] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.
- [60] A. S. Holevo. Bounds for the quantity of information transmitted by a quantum communication channel. *Problems of Information Transmission*, 9:177–183, 1973.
- [61] R. Horodecki, P. Horodecki, M. Horodecki, and K. Horodecki. Quantum entanglement. *Reviews of Modern Physics*, 81(2):865, 2009.
- [62] I. D. Ivanovic. How to differentiate between non-orthogonal states. *Physics Lett. A*, 123 (6):257–259, 1987.
- [63] M. Jakobi, C. Simon, N. Gisin, J. D. Bancal, C. Branciard, N. Walenta, and H. Zbinden. Practical private database queries based on a quantum-key-distribution protocol. *Phys. Rev. A*, 83 (2):022301, 2011.
- [64] D. Kahn. *The codebreakers: the comprehensive history of secret communication from ancient times to the internet*. Scribner, 1996.
- [65] J. Kaniewski. Self-testing of binary observables based on commutation. *Phys. Rev. A*, 95 (6):062323, 2017.
- [66] J. Kaniewski and S. Wehner. Device-independent two-party cryptography secure against sequential attacks. *New J. Phys.*, 81:055004, 2016.
- [67] N. Karimi. Optimal unambiguous discrimination of pure quantum states using SDP method. *Chinese Journal of Physics*, 72:681–687, 2021.
- [68] A. Kent. Unconditionally secure bit commitment. *Physical Review Letters*, 83:1447–1450, 1999.
- [69] I. Kerenidis, M. Laurière, F. L. Gall, and M. Rennela. Information cost of quantum communication protocols. *Quantum Info. Comput.*, 16 (3& 4):181–196, 2016.
- [70] J. Kilian. Founding Cryptography on Oblivious Transfer. In *Proc. 20th ACM STOC 1988*, pages 20–31, 1988.

- [71] W. Y. Kon and C. C. W. Lim. Provably-secure symmetric private information retrieval with quantum cryptography. *Entropy*, 23(1):54, 2021.
- [72] R. König, S. Wehner, and J. Wullschleger. Unconditional security from noisy quantum storage. *IEEE Transactions on Information Theory*, 58(3):1962–1984, 2012.
- [73] S. Kundu, J. Sikora, and E. Y-Z. Tan. A device-independent protocol for XOR oblivious transfer. *Quantum*, 6:725, 2022.
- [74] H. K. Lo. Insecurity of quantum secure computations. *Phys. Rev. A*, 56(2), 1997.
- [75] H. K. Lo and H. F. Chau. Is quantum bit commitment really possible? *Physical Review Letters*, 78(17):3410–3413, 1997.
- [76] T. Lunghi, J. Kaniewski, F. Bussi eres, R. Houlmann, M. Tomamichel, S. Wehner, and H. Zbinden. Practical Relativistic Bit Commitment. *Phys. Rev. Lett.*, 115:030502, 2015.
- [77] A. Maitra, G. Paul, and S. Roy. Device-independent quantum private query. *Phys. Rev. A*, 95 (4):042344, 2017.
- [78] L. Mancinska. Maximally entangled state in pseudo-telepathy games. *Computing with New Resources, Lecture Notes in Computer Science*, 8808:200–207, 2014.
- [79] D. Mayers. Unconditionally secure quantum bit commitment is impossible. *Physical Review Letters*, 78(17):3414–3417, 1997.
- [80] D. Mayers and A. Yao. Quantum cryptography with imperfect apparatus. In *In 39th Annual Symposium on Foundations of Computer Science FOCS 1998*, page 503–509, 1998.
- [81] D. Mayers and A. Yao. Self testing quantum apparatus. *Quantum Info. Comput.*, 4(4):273–286, 2004.
- [82] W. McCutcheon, A. Pappa, B. A. Bell, A. McMillan, A. Chailloux, T. Lawson, and M. S. Tame. Experimental verification of multipartite entanglement in quantum networks. *Nature Commun.*, 7:13251, 2016.
- [83] R. Merkle and M. Hellman. Hiding information and signatures in trapdoor knapsacks. *IEEE Trans. Inf. Theory*, 24(5):525–530, 2006.
- [84] M. Naor. Bit commitment using pseudorandomness. *J. Cryptology*, 4:151–158, 1991.
- [85] M. Naor and B. Pinkas. Distributed oblivious transfer. In *In Advances in Cryptology–ASIACRYPT 2000*, page 205–219, 2000.

- [86] M. A. Nielsen and I. L. Chuang. *Quantum computation and quantum information*. Cambridge University Press, 2010.
- [87] T. G. Noh. Counterfactual quantum cryptography. *Phys. Rev. Lett.*, 103(23):230501, 2009.
- [88] L. Olejnik. Secure quantum private information retrieval using phase-encoded queries. *Phys. Rev. A*, 84(2):022313, 2011.
- [89] A. Pappa, A. Chailloux, S. Wehner, E. Diamanti, and I. Kerenidis. Multipartite entanglement verification resistant against dishonest parties. *Phys. Rev. Lett.*, 108:260502, 2012.
- [90] A. Peres and D. R. Terno. Optimal distinction between non-orthogonal quantum states. *J. Phys. A: Math. Gen.*, 31:7105, 1998.
- [91] MV. P. Rao and M. Jakobi. Towards communication-efficient quantum oblivious key distribution. *Phys. Rev. A*, 87(1):012331, 2013.
- [92] B. Reichardt, F. Unger, and U. Vazirani. A classical leash for a quantum system: Command of quantum systems via rigidity of CHSH games. *Nature*, 496(7446):456, 2013.
- [93] R. Renner. Security of quantum key distribution. *International Journal of Quantum Information*, 6(01):1–127, 2008.
- [94] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.
- [95] T. Rudolph, R. W. Spekkens, and P. S. Turner. Unambiguous discrimination of mixed states. *Physical Review A*, 68(1):010301, 2003.
- [96] V. Scarani, A. Acín, G. Ribordy, and N. Gisin. Quantum cryptography protocols robust against photon number splitting attacks for weak laser pulse implementations. *Phys. Rev. Lett.*, 92:057901, 2004.
- [97] C. Schaffner, B. M. Terhal, and S. Wehner. Robust cryptography in the noisy-quantum-storage model. *Quantum Information and Computation*, 9(11 & 12):963–996, 2009.
- [98] A. Shamir. A polynomial time algorithm for breaking the basic Merkle-Hellman cryptosystem. In *In 23rd Annual Symposium on Foundations of Computer Science (FOCS 1982)*, pages 145–152, 1982.
- [99] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, 1948.
- [100] C. E. Shannon. Communication theory of secrecy systems. *The Bell System Technical Journal*, 28:656–715, 1949.

- [101] E. Shimon, O. Goldreich, and A. Lempel. A randomized protocol for signing contracts. *Communications of the ACM*, 28 (6):637–647, 1985.
- [102] P. W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *In IEEE Symposium on Foundations of Computer Science FOCS 1994*, pages 124–134, 1994.
- [103] P. W. Shor and J. Preskill. Simple proof of security of the BB84 quantum key distribution protocol. *Phys. Rev. Lett.*, 85:441–444, 2000.
- [104] J. Silman, A. Chailloux, N. Aharon, I. Kerenidis, S. Pironio, and S. Massar. Fully Distrustful Quantum Bit Commitment and Coin Flipping. *Phys. Rev. Lett.*, 106:220501, 2011.
- [105] S. Song and M. Hayashi. Quantum private information retrieval for quantum messages. In *IEEE International Symposium on Information Theory (ISIT) Melbourne, Australia*, pages 1052–1057, 2021.
- [106] A. Tavakoli, M. Smania, T. Vértesi, N. Brunner, and M. Bourennane. Self-testing non-projective quantum measurements in prepare-and-measure experiments. *Science Advances*, 6:16, 2020.
- [107] A. Vakhitov, V. Makarov, and D. R. Hjelle. Large pulse attack as a method of conventional optical eavesdropping in quantum cryptography. *Journal of Modern Optics*, 48(13):2023–2038, 2001.
- [108] U. Vazirani and T. Vidick. Certifiable quantum dice: or, true random number generation secure against quantum adversaries. In *Proc. 44th ACM STOC 2012*, pages 61–76, 2012.
- [109] U. Vazirani and T. Vidick. Fully device-independent quantum key distribution. *Phys. Rev. Lett.*, 113(14):140501, 2014.
- [110] H. P. Wang and R. G. Zhou. Multi-user quantum private query using symmetric multi-particle W state. *Int. J. Theor. Phys.*, 61:71, 2022.
- [111] S. Wehner, C. Schaffner, and B. M. Terhal. Cryptography from noisy storage. *Physical Review Letters*, 100(22):220502, 2008.
- [112] S. Wehner and J. Wullschleger. Composable security in the bounded-quantum-storage model. In *International Colloquium on Automata, Languages, and Programming ICALP 2008*, page 604–615, 2008.
- [113] C. Y. Wei, F. Gao, Q. Y. Wen, and T. Y. Wang. Practical quantum private query of blocks based on unbalanced-state Bennett-Brassard-1984 quantum-key-distribution protocol. *Sci. Rep.*, 4:7537, 2014.
- [114] S. Wiesner. Conjugate coding. *ACM SIGACT News*, 15(1):78–88, 1983.

- [115] L. Xiao, G.L. Long, F.G. Deng, and J.W. Pan. Efficient multiparty quantum-secret-sharing schemes. *Phys. Rev. A*, 69(5):052307, 2004.
- [116] H. Yang and M. Xiao. Multi-user quantum private query. *Quant. Inf. Proc.*, 19:253, 2020.
- [117] Y. G. Yang, S. J. Sun, P. Xu, and J. Tiang. Flexible protocol for quantum private query based on B92 protocol. *Quant. Info. Proc.*, 13:805, 2014.
- [118] A. C. Yao. Protocols for secure computations. In *In 23rd Annual Symposium on Foundations of Computer Science FOCS 1982*, pages 160–164, 1982.
- [119] T. Y. Ye, H. K. Li, and J. L. Hu. Multi-user quantum private query protocol. *Int. J. Theor. Phys.*, 59:2867–2874, 2020.
- [120] J. L. Zhang, F. Z. Guo, F. Gao, B. Liu, and Q. Y. Wen. Private database queries based on counterfactual quantum key distribution. *Phys. Rev. A*, 88(2):022334, 2013.
- [121] D. Zhu, L. Wang, and H. Zhu. Cryptanalysis of multi-user quantum private query protocol. *Int. J. Theor. Phys.*, 60:284–292, 2021.