A DEVICE INDEPENDENT QUANTUM KEY DISTRIBUTION APPROACH FOR THREE PARTY SYSTEM

A thesis submitted in partial fulfillment of the requirements for the award of the degree of

M.Tech

in

Computer Science and Engineering

By

VENUGOPAL S (206123032)

COMPUTER SCIENCE AND ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY
TIRUCHIRAPPALLI – 620015

DEC 2024

To Mother Nature

To 2025 - The International Year of Quantum Science and Technology (IYQ)

BONAFIDE CERTIFICATE

This is to certify that the project titled **A DEVICE INDEPENDENT QUANTUM KEY DISTRIBUTION APPROACH FOR THREE PARTY SYSTEM** is a bonafide record of the work done by

VENUGOPAL S (206123032)

in partial fulfillment of the requirements for the award of the degree of **Master of Technology** in **Computer Science and Engineering** of the **NATIONAL INSTITUTE OF TECHNOLOGY, TIRUCHIRAPPALLI**, during the year 2024-25.

Dr. KUNWAR SINGH
Guide

Dr. KUNWAR SINGH
Head of the Department

Project Viva-voce held on _____

Internal Examiner

External Examiner

ABSTRACT

In the Quantum Cryptography field, the CHSH (Clauser-Horne-Shimony-Holt) game plays a vital role in determining whether a system is in classical or quantum domain. This quantum advantage property is applied in determining the security of the quantum systems, especially in quantum cryptography, in the form of Device-Independent (DI) secure Quantum Key Distribution (QKD). The traditional CHSH game is designed for a system with two players, where each player has access to one entangled qubit from the Bell's state. In 2014, Vazirani and Vidick exploited the quantum advantage property from the CHSH game and proposed the first fully DIQKD protocol (FDIQKD) for two parties using two qubits. In 2019, Jyotirmoy Basak, Arpita Maitra and Subhamoy Maitra exploited the Multi-Party Pseudo-Telepathy (MPPT) game or GHZ game and proposed the DIQKD protocol for two parties using three qubits. In 2024, Hamza Jaffali and Frédéric Holweck proposed a new three-qubit CHSH game by extending the traditional CHSH game to three players using the three-qubit Greenberger–Horne–Zeilinger (GHZ) state and W state. In this research, we have proposed two approaches to handle the QKD scenario for a system involving three parties with DI security.

In our proposed work 1, which is based on the FDIQKD protocol, our contributions are three fold. First, we present our findings of new CHSH quantum games with quantum advantage in the four-qubit and five-qubit systems. These CHSH quantum games, when played using their quantum strategy, are found to outperform any classical strategy. Second, we apply the quantum advantage property of the existing three-qubit game to develop the corresponding three party DIQKD protocol. Third, we then apply the quantum advantage property of the newly developed four-qubit and five-qubit games to develop their corresponding four party and five party DIQKD protocols.

In our proposed work 2, which is based on the MPPT protocol, our contributions are two fold. First, we propose a modified MPPT approach for a new game since the existing MPPT could not give accurate results for the three party scenario. Second, we exploit this game to ensure the device is DI secure and then proceed with the key generation. We have provided the algorithms and detailed procedures for these respective approaches.

Keywords : Quantum Key Distribution, Device-Independent, CHSH Game, GHZ Game, Entanglement, Quantum Advantage, Quantum Games, Multi-Party Pseudo Telepathy

ACKNOWLEDGEMENTS

I would like to thank the following people for their support and guidance without whom the completion of this project in fruition would not be possible.

Dr. KUNWAR SINGH, Associate Professor and Head, Department of Computer Science and Engineering, for helping and guiding me throughout the course of this project. Without his guidance, I would not have been able to successfully complete this project. His patience and genial attitude is and always will be a source of inspiration to me.

Dr. S. SELVAKUMAR , Dr. S. JAYA NIRMALA , Dr. R. BALA KRISHNAN, the internal reviewers, for their insight and advice provided during the review sessions.

I am also thankful to the faculty and staff members of the Department of Computer Science and Engineering, my parents and my friends for their constant support and help.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Quantum computing is a new way of performing the computations based on the laws of nature. This is just a present reality of what Richard P Feynman, an eminent physicist and Nobel Laureate, had conceptualized some of its ideas several decades back itself. Compared to the classical computing, here we use qubits instead of bits, quantum gates instead of logic gates and an important paradigm called the 'measurement'. This measurement phenomenon is what collapses the multiple probabilities of all states into a single definite value which we can measure in the physical world. The goal of the quantum computing is to work with these probabilities for our advantage on the desired measurement. Also in quantum computing we can apply the concepts of superposition and entanglement on qubits, which provide a unique way of performing computations.

Superposition [1, 2] is a phenomena of quantum theory which describes the ability of a quantum system to act as if it is in multiple states at the same time until it is measured. The most famous thought process of understanding the superposition principle in quantum mechanics is through the double-slit experiment. Quantum mechanics uses probability waves to represent the states of particles. When quantum particles are acting according to probability waves, they are said to be in a state of quantum superposition.

Entanglement [1, 2] is a phenomena of quantum theory where a group of particles exist in such a way that the state of each particle of that group cannot be described independently of the state of the others even though they are separated by a large distance. It was described by Einstein in his famous quote as the "spooky action at a distance".

Quantum games are the games which are played by applying the principles of quantum mechanics (entanglement, superposition etc.) and quantum resources (qubits, quantum gates etc.). Some of these quantum games could even outperform the best classical strategy possible, the phenomenon known as the quantum advantage. This quantum advantage phenomenon is very crucial in the field of quantum cryptography in detecting the presence of unwanted adversaries.

Cryptography helps to protect the data from access by unwanted adversaries. Usually this security assumption is provided by the intractable behaviour of any proven hardness problem in the classical world. But recent developments in quantum algorithms, like the Shor's algorithm, can break these hardness and make them

tractable, thus posing a great threat to the present cryptographic methods. Quantum cryptography applies the quantum mechanical principles into the cryptographic algorithms.The security assumption in quantum cryptography is provided by the laws of nature such as the No Cloning Theorem, which are not breakable, and thus provides higher level of security than classical cryptography.

Key Distribution is a cryptographic primitive which enables two parties to securely generate a common secret key over an insecure network. After sharing a secret key, both parties can make secure communication using an efficient symmetric cryptosystem. In the literature, most of the key distribution protocols are based on the hardness of prime factorization and discrete logarithm problem.

In 1994, Peter Shor showed that prime factorization and discrete logarithm problem can be solved in polynomial time on a quantum computer. In other words, once a quantum computer becomes a reality all the existing Key Distribution algorithm will be broken in a few seconds. Quantum Key Distribution (QKD) emerges as a promising alternative for secure key sharing in the quantum era. By exploiting the principles of quantum mechanics, such as entanglement, QKD establishes provably secure keys between parties. This is very critical because in future an attacker with a powerful quantum computer could break the currently existing cryptographic algorithms like RSA, Elliptic curve etc. So we have to design new cryptographic algorithms which will be quantum-resistant to these futuristic attacks.

The first working QKD algorithm was proposed in 1984 by Bennett and Brassard which was named as the BB84 protocol [3, 4]. Based on the prepare and measure model, the BB84 did not use entanglement and it can handle two parties. Then E91 protocol [5] was proposed by Ekert in 1991 using the entanglement property, thus paving a new path called the entanglement-based models. Later, further variations of these QKD protocols with improvements were proposed in the upcoming years. Some of the significant protocols of them would be the B92 and BBM92. Also with further cryptanalysis of these protocols, various attacks on these protocols were also discovered like the Photon Number Splitting (PNS) attack. This leads to the definition of new security standards for developing the QKD protocols. One of such security standards is the Device Independent (DI) security.

The CHSH game is one of such quantum games with a clear quantum advantage and widely used in various domains. It uses a pair of entangled states, also known as Bell's state or EPR state [6]. But it can handle only for a system involving two parties where each party holds one entangled qubit. It has been proved that by using the classical strategy, the maximum attainable success probability for the CHSH game is 75%. On the other hand, by using the quantum strategy, the maximum attainable success probability for the same CHSH game is around 85%

[7].

Later in 2003, Gilles Brassard, Anne Broadbent, and Alain Tapp [3] proposed another quantum game called the Multi-Party Pseudo-Telepathy (MPPT). In case of a quantum protocol, the MPPT provides a perfect success whereas in case of a classical protocol, it does not.

Quantum key distribution (QKD), the quantum analogous of the classical key distribution, satisfies this requirement of creating a common and secure shared key which is resistant to attacks even from a quantum computer. Real world QKD applications are performed through polarized photons which requires some prerequisite setup of physical devices. The quantum computers which are deployed at present suffer from serious physical implementation errors. So its possible for an attacker to exploit this drawback to his own advantage by performing some malicious operations unknown to the QKD users by using these errors. Thus we should always assume our devices as imperfect or even malicious (i.e) the device which we use should never be trusted blindly. This notion of security is called Device Independent (DI). DI security is very essential in the present era of QKD to provide the highest level of security.

In 2014, Vazirani and Vidick [8] exploited the quantum advantage property from the CHSH game and proposed the DIQKD protocol for two parties using two qubits. A fully DIQKD proptocol (FDIQKD) was designed by exploiting the quantum advantage from the traditional CHSH game. This FDIQKD protocol broadly consists of two phases - Testing phase and Key Genration phase. During Testing phase, the players play the CHSH game and check if they are able to achieve quantum advantage or not. If not, then it denotes a presence of some adversarial behaviour in the system or device, which could compromise the security of their communication. So, if the testing phase fails, they will abort the protocol, else, they will proceed to the Key Genration phase. Quantum advantage is achieved only when all the devices involved are honest.

In 2019, Jyotirmoy Basak, Arpita Maitra and Subhamoy Maitra [9] exploited the MPPT game or GHZ game and proposed the DIQKD protocol for two parties using three qubits. The GHZ game has been used to check for the quantum advantage. Since the GHZ game gives a perfect success, this MPPTQKD was more efficient than the previous FDIQKD. But both these FDIQKD and MPPTQKD can only work for a two party system.

In [10], a three qubit entangled state (either a GHZ or W state) is used where each of the three player will manipulate his/her qubit by applying their specific unitary transformations. Any unitary operator which can be applied on a single qubit can be defined with the help of three angles. These three angles will rotate the qubits along the x, y and z-axes respectively. They extended the traditional

CHSH game conditions to fit the three qubit system and also provided the strategy to achieve the quantum advantage but they did not give any strategy for applying it to the QKD domain.

Real world systems cannot be constrained with limited use of the quantity of qubits. Hence it is very critical to discover and develop new quantum games which can handle large systems involving many qubits. One of such key areas where this concept plays a vital role is quantum cryptography wherein, for a system involving N parties we have to share a common key among all of them, a concept known as the Quantum Key Distribution (QKD). Thus the real world applications of these multi-qubit quantum games are enormous as well as essential in various fields.

Not all quantum games are found to provide quantum advantage. Some may and some may not. So its crucial to distinguish the quantum games with clear quantum advantage from the unclear ones. For each system involving M number of qubits, we can obviously come up with several number of classical strategies and quantum strategies, and this count keeps on increasing to a very high value as the value of M increases. This in turn increases the complexity involved and the search space from where we may have to find out the specific one with a clear quantum advantage.

In this thesis, our contributions are as follows. In Chapter 2, we deal with the preliminaries required for the understanding of the research. In Chapter 3, we present the literature survey for our work. In Chapter 4, first, we present our findings of new CHSH quantum games with quantum advantage in the four-qubit and five-qubit systems. These CHSH quantum games, when played using their quantum strategy, are found to outperform any classical strategy. Second, we apply the quantum advantage property of the existing three-qubit game to develop the corresponding three party DIQKD protocol. Third, we then apply the quantum advantage property of the newly developed four-qubit and five-qubit games to develop their corresponding four party and five party DIQKD protocols. In Chapter 5, we provide another different approach to perform the three party DIQKD protocol. In Chapter 6, we provide the conclusions and future directions to follow up from this research. In Chapter 7, we present our publication outcomes from this research. Finally, we conclude the thesis with the code attachments and the bibliography.

CHAPTER 2

PRELIMINARIES

2.1 The Game Environment

The game environment consists of several participating players and one referee. Let the number of players involved be N. The players are identified by P_1, P_2, till P_N. The referee is identified by R. The players are allowed to communicate among themselves and finalize the strategy to be used by each one of them before the start of the game. Once the game starts, the players cannot communicate among themselves until the game ends. The referee creates a random string X consisting of binary digits 0's and 1's of length N. The digit at the i^{th} position X_i is shared with the player P_i as the input. The player P_i will then apply his/her strategy based on the input X_i and gets a result A_i, which is returned back as his/her output to the referee. The referee collects all such outputs A_1, A_2, till A_N and checks against the winning condition of the game which can be formulated as Eqn.(2.1) where f denotes the function of all the input values X_i and g denotes the function of all the output values A_i. Fig 2.3 depicts the game environment.

$$f(X_1, X_2, ..., X_N) = g(A_1, A_2, ..., A_N) \tag{2.1}$$

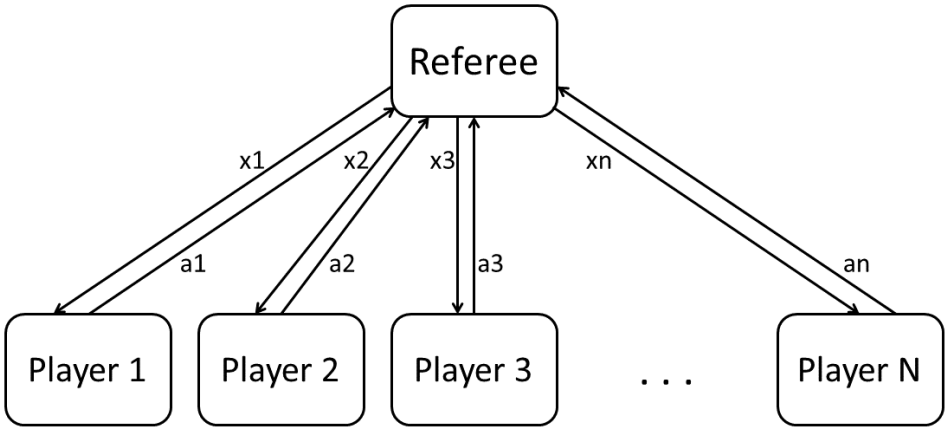

Figure 2.1: Multi-Qubit CHSH Game involving N Parties

2.2 Two Qubit CHSH Game

The CHSH game is one of the most important concepts in quantum computing since it enables us to realize the violation of the Bell's inequality in practice. The game consists of two parties Alice and Bob who are separated very far apart and they cannot communicate among themselves. While the maximum success probability using classical strategy is 75%, the quantum strategy, by using two qubits in an entangled state (EPR pair or Bell state)[11], increases the success probability to around 85%. Only the people with access to the quantum resources will be able to achieve a higher success rate than their classical counter-part. Hence the traditional CHSH game clearly defines the quantum advantage in a two qubit system.

The two parties involved Alice and Bob can discuss and pick a strategy before the game starts, but they can't communicate after the game begins. Th strategy for one party is independent of the other party and so Alice and Bob can manipulate their own input however they wish, according to their strategy, and produce an output.

Let x $\in \{0,1\}$ and y $\in \{0,1\}$ denote the inputs given by referee R to Alice and Bob respectively. Let a $\in \{0,1\}$ and b $\in \{0,1\}$ be the outputs produced by Alice and Bob respectively. The winning condition for CHSH game is given in Eqn.(2.2)

$$x \wedge y = a \oplus b \tag{2.2}$$

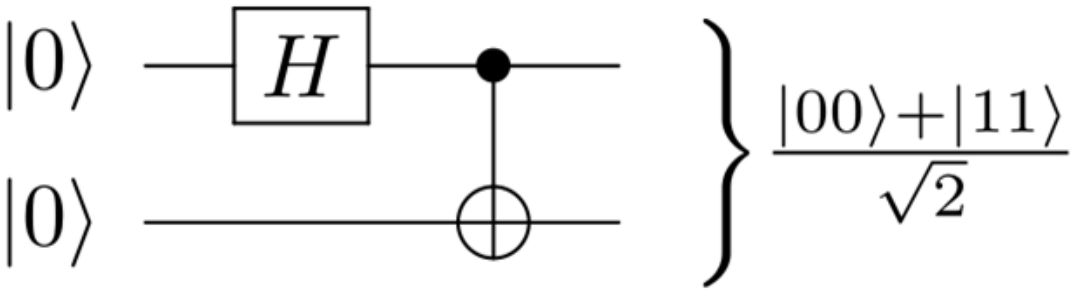

Figure 2.2: CHSH Game Setup for Two Parties (From [12])

2.3 Three Qubit CHSH Game

Similar to the traditional CHSH game which uses the Bell's state of two entangled qubits, the three Qubit CHSH Game [10] is designed using three entangled qubits where every party will hold one of them each. An unitary operator can be defined as the operator which rotates the qubit vector around the bloch sphere along the three axes (x, y and z axes). Any unitary operator used in quantum domain can be described as the rotation along the three axes.

Here, we use the GHZ state = $\frac{1}{\sqrt{2}}(|0^3\rangle + |1^3\rangle)$. After that each of the qubit is shared to each party. Each party will then apply his own defined unitary transformation based on the input recieved from the referee R. Let x $\in$ {0,1}, y $\in$ {0,1} and z $\in$ {0,1}denote the inputs given by referee R to Alice, Bob and Chuck respectively. Let a $\in$ {0,1}, b $\in$ {0,1} and c $\in$ {0,1} be the outputs produced by Alice, Bob and Chuck respectively. Let $U_{1,x}$ be the unitary transformation applied by Alice on seeing the input x. Similarly let $U_{2,y}$ and $U_{3,z}$ be the unitary transformations for Bob and Chuck respectively. For each state of their input x, y and z, the parties will apply the predefined unitary transformation strategy and then measure in the computational basis. These results a, b and c are then sent to referee R who will check against that spcific three Qubit CHSH Game winning condition and sees if it exhibits a quantum advantage or not. Th maximum quantum advantage achieveable in three qubit CHSH game is same as the two qubit CHSH game (i.e) around 85%.

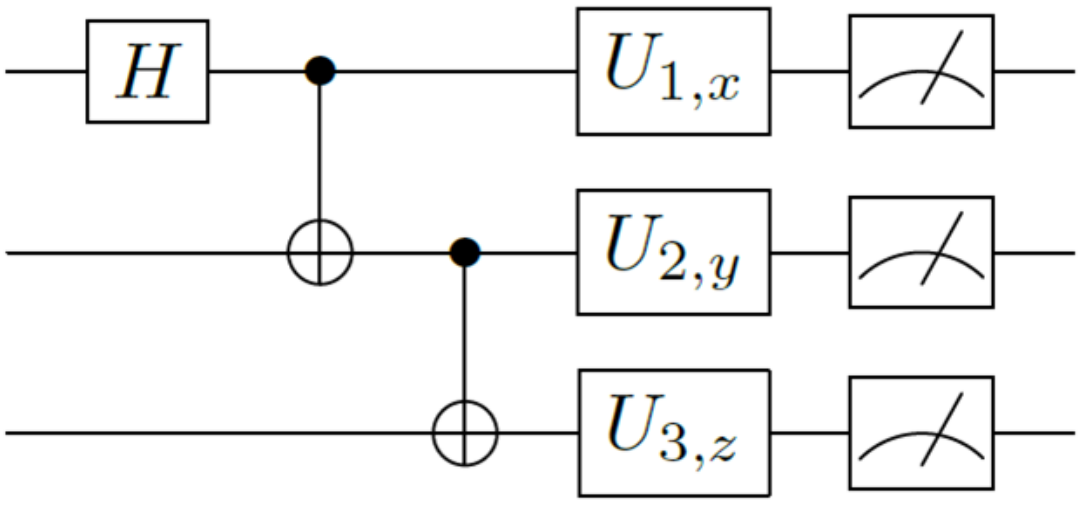

Figure 2.3: CHSH Game Setup for Three Parties

2.4 GHZ Game (or) MPPT Game

While the CHSH game gives a maximum success of around 85%, the GHZ game [13] gives a perfect success rate of 100% using the quantum strategy. While the CHSH game uses the two qubit entangled Bell state, this game uses the Greenberger–Horne–Zeilinger (GHZ) state which is the entanglement of three qubits. Also the GHZ game can handle 'n' number of parties where n¿=3. Hence its also called as the Multi-Party Pseudo Telepathy (MPPT) game.

A detailed description of the game is given below.

For any $n \geq 3$, game G_n consists of n players. Each player A_i receives a single input bit x_i and is requested to produce a single output bit y_i. The players are promised that the number of 1's among their inputs is even (i.e) $\sum x_i \equiv 0 \pmod 2$. We require that,

$$\sum_{i}^{n} y_i \equiv \frac{1}{2} \sum_{i}^{n} x_i \pmod 2 \tag{2.3}$$

provided $\sum x_i \equiv 0 \pmod 2$. We say that $x = x_1x_2...x_n$ is the question and $y = y_1y_2...y_n$ is the answer.

If the n players are allowed to share prior entanglement, then they can always win game G_n. We define the following n-qubit entangled quantum states $|\phi_n{}^+\rangle$ and $|\phi_n{}^-\rangle$.

$$|\phi_n{}^+\rangle = \tfrac{1}{\sqrt{2}}(|0^n\rangle + |1^n\rangle)$$
$$|\phi_n{}^-\rangle = \tfrac{1}{\sqrt{2}}(|0^n\rangle - |1^n\rangle)$$

Let H denote the Hadamard transformation.

$$H|0\rangle = \tfrac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$
$$H|1\rangle = \tfrac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

Let S denote the unitary transformation defines as below.

$$S|0\rangle = |0\rangle$$
$$S|1\rangle = i|1\rangle$$

For our three party DIQKD scenario, we can see that if S is applied to any two qubits of $|\phi_n^+\rangle$, then the resulting state is $|\phi_n^-\rangle$. Also we can see the same in reverse that if S is applied to any two qubits of $|\phi_n^-\rangle$, then the resulting state is $|\phi_n^+\rangle$.

From the above calculations we can derive that if the qubits of $|\phi_n^+\rangle$ are distributed among the n players, and if exactly m of them apply S to their qubit, the resulting global state will be $|\phi_n^+\rangle$ if $m \equiv 0$ (mod 4) and $|\phi_n^-\rangle$ if $m \equiv 0$ (mod 2).

The referee will create the input string $x = x_1x_2...x_n$ in such a way that the sum of all imputs is even (i.e) $\sum x_i \equiv 0$ (mod 2). So the referee has only two options to generate the input string, either $\frac{1}{2}\sum_i^n x_i$ (mod 2) $\equiv 0$ or $\frac{1}{2}\sum_i^n x_i$ (mod 2) $\equiv 1$.

After applying the quantum strategy, we can see that $\sum_i^n y_i$ is even if $\frac{1}{2}\sum_i^n x_i$ (mod 2) $\equiv 0$ and odd if $\frac{1}{2}\sum_i^n x_i$ (mod 2) $\equiv 1$.

Thus, in both the cases, the Eqn.(2.3) is always satisfied.

CHAPTER 3

PROPOSED WORK 1

In the Quantum Computing field, the CHSH (Clauser-Horne-Shimony-Holt) game plays a vital role in determining whether a system is in classical or quantum domain. This quantum advantage property is applied in determining the security of the quantum systems, especially in quantum cryptography, in the form of Device-Independent (DI) secure Quantum Key Distribution (QKD). The traditional CHSH game is designed for a system with two players, where each player has access to one entangled qubit from the Bell's state. In 2024, Hamza Jaffali and Frédéric Holweck proposed a new three-qubit CHSH game [10] by extending the traditional CHSH game to three players using the three-qubit Greenberger–Horne–Zeilinger (GHZ) state and W state. Based on [10], we have designed for four and five qubits.

3.1 Four Qubit Game

3.1.1 Preparing the entangled state

Before the game begins, an extended GHZ entangled state of four qubits is prepared (or) received from a third party. The resultant qubits are assumed to be maximally entangled among each other. Then, the qubits are shared with the players as one qubit for each player.

In the four qubit system, we have identified two games which yield quantum advantage and so the parties could use any one of these conditions to check for the game's success in the quantum domain.

3.1.2 Winning Condition For Game 1

Here the LHS consists of the sum of all ${}_4C_2$ combinations of the inputs. The RHS is the XOR of all the four outputs A_i.

$$\begin{aligned}(X_1 \wedge X_2) \vee (X_1 \wedge X_3) \vee (X_1 \wedge X_4) \\ \vee (X_2 \wedge X_3) \vee (X_2 \wedge X_4) \vee (X_3 \wedge X_4) \\ = A_1 \oplus A_2 \oplus A_3 \oplus A_4 \end{aligned} \quad (3.1)$$

3.1.3 Winning Condition For Game 2

Here the LHS consists of the sum of all ${}_4C_3$ combinations of the inputs. The RHS is the XOR of all the four outputs A_i.

$$(X_1 \wedge X_2 \wedge X_3) \vee (X_1 \wedge X_2 \wedge X_4) \vee (X_1 \wedge X_3 \wedge X_4) \vee (X_2 \wedge X_3 \wedge X_4) = A_1 \oplus A_2 \oplus A_3 \oplus A_4 \quad (3.2)$$

3.2 Five Qubit Game

Here, all the procedure is same as that of the four qubit game, except that an extended GHZ entangled state of five qubits is prepared (or) received from a third party. The resultant qubits are assumed to be maximally entangled among each other.

3.2.1 Winning Condition For Game

Here the LHS consists of the sum of all ${}_5C_3$ combinations of the inputs. The RHS is the XOR of all the five outputs A_i.

$$\begin{aligned}(X_1 \wedge X_2 \wedge X_3) \vee (X_1 \wedge X_2 \wedge X_4) &\vee (X_1 \wedge X_2 \wedge X_5) \vee (X_1 \wedge X_3 \wedge X_4) \\ &\vee (X_1 \wedge X_3 \wedge X_5) \vee (X_1 \wedge X_4 \wedge X_5) \\ &\vee (X_2 \wedge X_3 \wedge X_4) \vee (X_2 \wedge X_3 \wedge X_5) \\ &\vee (X_2 \wedge X_4 \wedge X_5) \vee (X_3 \wedge X_4 \wedge X_5) \\ &= A_1 \oplus A_2 \oplus A_3 \oplus A_4 \oplus A_5 \quad (3.3)\end{aligned}$$

3.3 Simulation

3.3.1 Classical Strategy

The classical strategy for the games are simulated using python. Upon receiving the input, each player P_i has the liberty of choosing one of the four classical options to return as output (i.e) return a zero always ($A_i = 0$), return a one always ($A_i = 1$), return the same input as output ($A_i = X_i$) and return the complement of the input as output ($A_i = \tilde{X}_i$). Since each player P_i has four choices, the total number of classical outputs possible for the N qubit game is 4^N (i.e) 256 combinations for the four qubit game and 1024 combinations for the five qubit game.

Table 3.1: Quantum Strategy For Four Qubit Game 1

$(\theta,\ \phi,\ \lambda)$	Strategy 1	Strategy 2	Strategy 3	Strategy 4
$U_{1,\ 0}$	$(\frac{\pi}{2},\ \frac{13\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{30\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{46\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{61\pi}{32},\ \frac{\pi}{32})$
$U_{1,\ 1}$	$(\frac{\pi}{2},\ \frac{29\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{46\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{62\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{77\pi}{32},\ \frac{17\pi}{32})$
$U_{2,\ 0}$	$(\frac{\pi}{2},\ \frac{13\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{30\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{46\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{61\pi}{32},\ \frac{\pi}{32})$
$U_{2,\ 1}$	$(\frac{\pi}{2},\ \frac{29\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{46\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{62\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{77\pi}{32},\ \frac{17\pi}{32})$
$U_{3,\ 0}$	$(\frac{\pi}{2},\ \frac{13\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{30\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{46\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{61\pi}{32},\ \frac{\pi}{32})$
$U_{3,\ 1}$	$(\frac{\pi}{2},\ \frac{29\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{46\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{62\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{77\pi}{32},\ \frac{17\pi}{32})$
$U_{4,\ 0}$	$(\frac{\pi}{2},\ \frac{13\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{30\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{46\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{61\pi}{32},\ \frac{\pi}{32})$
$U_{4,\ 1}$	$(\frac{\pi}{2},\ \frac{29\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{46\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{62\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{77\pi}{32},\ \frac{17\pi}{32})$

Table 3.2: Quantum Strategy For Four Qubit Game 2

$(\theta,\ \phi,\ \lambda)$	Strategy 1	Strategy 2	Strategy 3	Strategy 4
$U_{1,\ 0}$	$(\frac{\pi}{2},\ \frac{10\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{26\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{42\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{58}{32},\ \frac{\pi}{32})$
$U_{1,\ 1}$	$(\frac{\pi}{2},\ \frac{26}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{42\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{58\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{74\pi}{32},\ \frac{17\pi}{32})$
$U_{2,\ 0}$	$(\frac{\pi}{2},\ \frac{10}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{26\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{42\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{58\pi}{32},\ \frac{\pi}{32})$
$U_{2,\ 1}$	$(\frac{\pi}{2},\ \frac{26}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{42\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{58\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{74\pi}{32},\ \frac{17\pi}{32})$
$U_{3,\ 0}$	$(\frac{\pi}{2},\ \frac{10\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{26\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{42\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{58\pi}{32},\ \frac{\pi}{32})$
$U_{3,\ 1}$	$(\frac{\pi}{2},\ \frac{26}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{42\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{58\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{74\pi}{32},\ \frac{17\pi}{32})$
$U_{4,\ 0}$	$(\frac{\pi}{2},\ \frac{10\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{26\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{42\pi}{32},\ \frac{\pi}{32})$	$(\frac{\pi}{2},\ \frac{58\pi}{32},\ \frac{\pi}{32})$
$U_{4,\ 1}$	$(\frac{\pi}{2},\ \frac{26\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{42\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{58\pi}{32},\ \frac{17\pi}{32})$	$(\frac{\pi}{2},\ \frac{74\pi}{32},\ \frac{17\pi}{32})$

Table 3.3: Quantum Strategy For Five Qubit Game

$(\theta{=}\frac{\pi}{2},\ \phi,\ \lambda)$	Strategy 1	Strategy 2	Strategy 3	Strategy 4	Strategy 5
$U_{1,\ 0}$	$(\frac{5\pi}{32},\ \frac{\pi}{32})$	$(\frac{26\pi}{32},\ \frac{\pi}{32})$	$(\frac{42\pi}{32},\ \frac{\pi}{32})$	$(\frac{58}{32},\ \frac{\pi}{32})$	$(\frac{58}{32},\ \frac{\pi}{32})$
$U_{1,\ 1}$	$(\frac{21}{32},\ \frac{17\pi}{32})$	$(\frac{42\pi}{32},\ \frac{17\pi}{32})$	$(\frac{58\pi}{32},\ \frac{17\pi}{32})$	$(\frac{74\pi}{32},\ \frac{17\pi}{32})$	$(\frac{74\pi}{32},\ \frac{17\pi}{32})$
$U_{2,\ 0}$	$(\frac{5}{32},\ \frac{\pi}{32})$	$(\frac{26\pi}{32},\ \frac{\pi}{32})$	$(\frac{42\pi}{32},\ \frac{\pi}{32})$	$(\frac{58\pi}{32},\ \frac{\pi}{32})$	$(\frac{58\pi}{32},\ \frac{\pi}{32})$
$U_{2,\ 1}$	$(\frac{26}{32},\ \frac{17\pi}{32})$	$(\frac{42\pi}{32},\ \frac{17\pi}{32})$	$(\frac{58\pi}{32},\ \frac{17\pi}{32})$	$(\frac{74\pi}{32},\ \frac{17\pi}{32})$	$(\frac{74\pi}{32},\ \frac{17\pi}{32})$
$U_{3,\ 0}$	$(\frac{5\pi}{32},\ \frac{\pi}{32})$	$(\frac{26\pi}{32},\ \frac{\pi}{32})$	$(\frac{42\pi}{32},\ \frac{\pi}{32})$	$(\frac{58\pi}{32},\ \frac{\pi}{32})$	$(\frac{58\pi}{32},\ \frac{\pi}{32})$
$U_{3,\ 0}$	$(\frac{5}{32},\ \frac{17\pi}{32})$	$(\frac{42\pi}{32},\ \frac{17\pi}{32})$	$(\frac{58\pi}{32},\ \frac{17\pi}{32})$	$(\frac{74\pi}{32},\ \frac{17\pi}{32})$	$(\frac{74\pi}{32},\ \frac{17\pi}{32})$
$U_{4,\ 0}$	$(\frac{5\pi}{32},\ \frac{\pi}{32})$	$(\frac{26\pi}{32},\ \frac{\pi}{32})$	$(\frac{42\pi}{32},\ \frac{\pi}{32})$	$(\frac{58\pi}{32},\ \frac{\pi}{32})$	$(\frac{58\pi}{32},\ \frac{\pi}{32})$
$U_{4,\ 1}$	$(\frac{26\pi}{32},\ \frac{17\pi}{32})$	$(\frac{42\pi}{32},\ \frac{17\pi}{32})$	$(\frac{58\pi}{32},\ \frac{17\pi}{32})$	$(\frac{74\pi}{32},\ \frac{17\pi}{32})$	$(\frac{74\pi}{32},\ \frac{17\pi}{32})$
$U_{5,\ 0}$	$(\frac{5\pi}{32},\ \frac{\pi}{32})$	$(\frac{5\pi}{32},\ \frac{\pi}{32})$	$(\frac{5\pi}{32},\ \frac{\pi}{32})$	$(\frac{5\pi}{32},\ \frac{\pi}{32})$	$(\frac{5\pi}{32},\ \frac{\pi}{32})$
$U_{5,\ 1}$	$(\frac{26\pi}{32},\ \frac{17\pi}{32})$	$(\frac{26\pi}{32},\ \frac{17\pi}{32})$	$(\frac{26\pi}{32},\ \frac{17\pi}{32})$	$(\frac{26\pi}{32},\ \frac{17\pi}{32})$	$(\frac{26\pi}{32},\ \frac{17\pi}{32})$

3.3.2 Quantum Strategy

The quantum strategy for the games are simulated using the IBM quantum computing framework - Qiskit. First, we create the extended GHZ entangled state of all the N qubits. Then, we share each of the resultant qubit to one per player. The qubits are considered to b maximally entangled. The player P_i applies his/her own unitary transformation on his/her qubit based on the input X_i received. Then the qubits are measured and the player gets an output A_i. This output A_i is shared to the referee R who will check for the winning condition based on the value N.

For finding the strategy with quantum advantage, we follow the below procedure. Each player P_i can receive one of the two options (0 or 1) as the input X_i. For each of the recieved input option, P_i can apply unitary transformations involving the rotations along the three axes. Thus each player P_i needs to have six defined values of angles, first set of three angles for X_i=0 and second set of three angles for X_i=1. Let $U_{i,\ X_i}$, consisting of three angles $\theta_{i,\ X_i}$, $\phi_{i,\ X_i}$ and $\lambda_{i,\ X_i}$ denoting the rotations along the X, Y and Z axes respectively in the bloch sphere, be the Unitary operator applied by the player P_i on his/her qubit when his/her input is X_i. Since X_i is binary, each player P_i has two unitary transformations to apply (i.e) $U_{i,\ 0}$ when input X_i=0 and $U_{i,\ 1}$ when input X_i=1.

Now we need to perform the exhaustive search across all possible values of θ, ϕ and λ. Going through all of these possibilities is a time consuming process and it also depends on the simulation factor s. This simulation factor s is similar to that of the learning factor used in Gradient Descent algorithm. If s is chosen large, we will not be able to capture the optimal parameters, also if s is chosen small, the convergence to optimal result may take a very long time.

Let t1, t2 and t3 denote the arbitrary values which vary by a small simulation factor s during every round of simulation run. The initial values are set at t1 = $\frac{\pi}{2}$, t2 = $\frac{\pi}{32}$ and t3 = $\frac{\pi}{32}$. The value of simulation factor s = $\frac{\pi}{32}$ and after each run, the values of t1, t2 and t3 get updated for the next simulation. The whole region of 2π is thus equally divided into 64 sectors.

In [10], some preset conditions have been established in determining these required parameters, like $\mid \lambda_{i,0} - \lambda_{i,1} \mid = \frac{\pi}{2}$ and $\theta_{i,j} = \frac{\pi}{2}$ which will be used here. Also we can see that all the players can apply the same quantum strategy to reach the maximum advantage. Let a1, a2 and a3 denote the values, derived from t1, t2 and t3, used by all the players P_i for a particular simulation round when input X_i=0. So, by the preset conditions, we have a1 = t1, a2 = t2 and a3 = t3. Let a4, a5 and a6 denote the values, derived from t1, t2 and t3, used by all the players P_i for a particular simulation round when input X_i=1. So, by the preset conditions, we have a4 = t1, a5 = t2 + $\frac{\pi}{2}$ and a6 = t3 + $\frac{\pi}{2}$. Thus, the rotation along the X axis always

remains a constant (i.e) $\theta_{i,j}=\frac{\pi}{2}$ for all the rounds. Only the values of ϕ and λ are changed by a factor s. Thus the total no. of combinations of search space involved can be calculated as = (64 sectors along Y axis) * (64 sectors along Z axis), which makes up to 4096.

The simulation factor here is set at s = $\frac{\pi}{32}$ = $\frac{2\pi}{64}$. Since we do it along two axes (Y axis and Z axis) for the search space, we get 64*64 = 4096 combinations. Suppose if we want to improve the learning rate further by diving into d sectors to gt more accurate results, then the simulation factor becomes s = $\frac{2\pi}{d}$, which makes up to d^2 combinations.

The final quantum circuit constructions for the four qubit game 1, four qubit game 2 and five qubit game are depicted in Fig 3.1, Fig 3.2, and Fig 3.3 respectively.

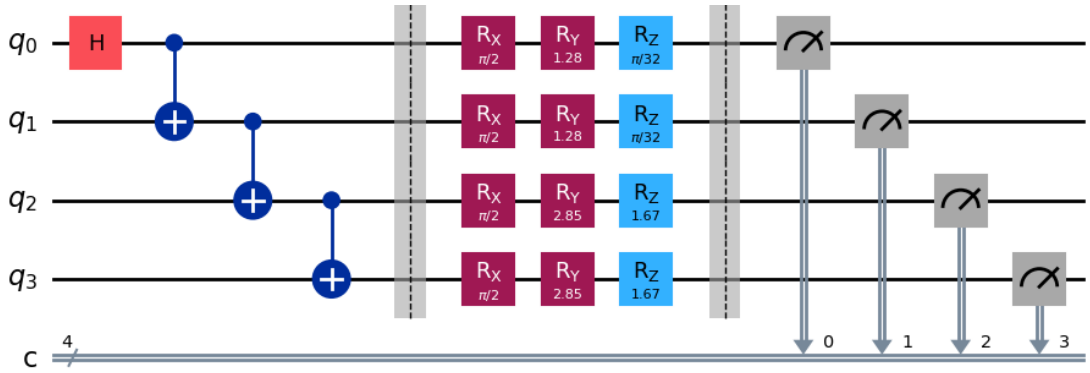

Figure 3.1: Four Qubit Circuit with the Quantum Strategy for Game 1 where (X1, X2, X3, X4) = (0, 0, 1, 1) and (t1, t2, t3) = ($\frac{\pi}{2}$, $\frac{13\pi}{32}$, $\frac{\pi}{32}$)

Figure 3.2: Four Qubit Circuit with the Quantum Strategy for Game 2 where (X1, X2, X3, X4) = (0, 0, 1, 1) and (t1, t2, t3) = ($\frac{\pi}{2}$, $\frac{10\pi}{32}$, $\frac{\pi}{32}$)

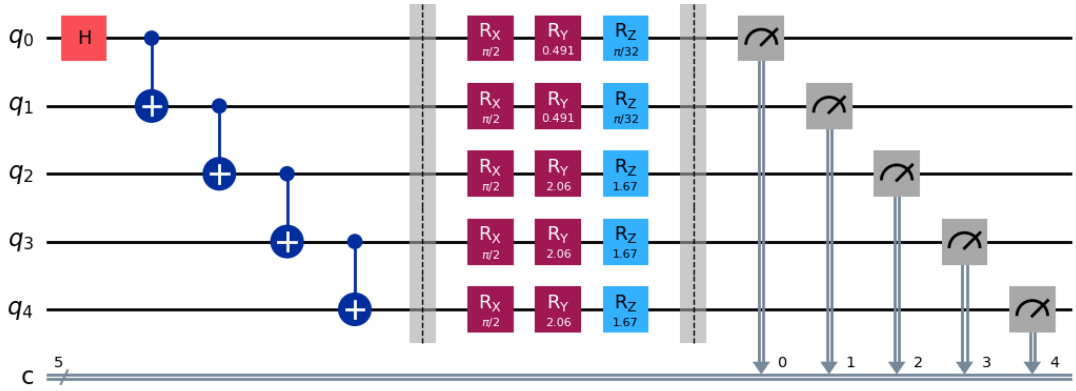

Figure 3.3: Five Qubit Circuit with the Quantum Strategy where (X1, X2, X3, X4, X5) = (0, 0, 1, 1, 1) and (t1, t2, t3) = ($\frac{\pi}{2}$, $\frac{5\pi}{32}$, $\frac{\pi}{32}$)

3.4 Results of Quantum Advantage

The results of the simulation using quantum strategy for each run, which gives the maximum achieved winning probability using a particular set of parameters a1, a2, a3, a4, a5 and a6, are plotted in a graph. The X-axis of the graph ranges from 0 to 4096, which denotes the total number of simulations possible for our chosen simulation factor s = $\frac{2\pi}{64}$ = $\frac{\pi}{32}$. The Y-axis of the graph ranges from 0 to 1, which denotes the maximum achieved winning probability using that specific quantum strategy.

We can find specific peaks in the graphs from Fig 3.4, Fig 3.5, and Fig 3.6 which reach to the traditional CHSH game maximum of around 85% (i.e) $\cos^2(\frac{\pi}{8})$. This clearly states the presence of quantum advantage in these games. Also we could observe that for each quantum strategy which uses N number of qubits, N number of strategies are possible since we can find N number of crests in the graph which reach to a maximum of about $\cos^2(\frac{\pi}{8})$. In addition, there are N troughs which reach to a minimum of about 1 - $\cos^2(\frac{\pi}{8})$. So if we modify the game conditions in (3.1), (3.2) and (3.3), by replacing A_1 in the RHS with its complement $\tilde{A}_1$, the graph gets inverted and so we get further N more strategies. Moreover, the variables ${}_4C_2$, ${}_4C_3$ and ${}_5C_3$ are quite interesting to observe because they lie around the central region of the Pascal's triangle.

In classical strategy, there are 256 combinations for the four qubit game and 1024 combinations for the five qubit game. Using classical strategy, the maximum achieved winning probability is not more than 70% in all the newly proposed three games.

Table 3.1, Table 3.2 and Table 3.3 list the parameters of the quantum strategies with quantum advantage for the four qubit game 1, four qubit game 2 and five qubit game respectively.

Through the above observed patterns, we propose the below conjectures :

- Conjecture 1 : Extension of this same simulation procedure to N qubit systems can lead to quantum advantage if the input combinations lie around the central region of the Pascal's triangle
- Conjecture 2 : In an N qubit system, there are atleast 2*N quantum strategies to achieve the quantum advantage.

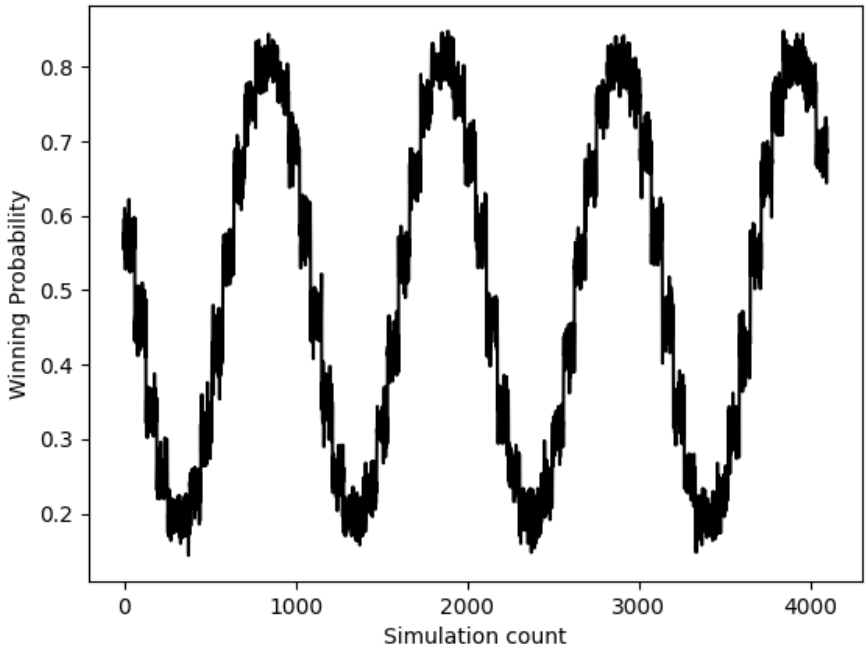

Figure 3.4: Plot of the winning probabilities for Four Qubit Game 1

3.5 Designing the DIQKD Protocols

Till now, we have clearly defined the four qubit and five qubit games which demonstrate quantum advantage from their classical part. Now we exploit this quantum advantage and propose their corresponding DI secure QKD protocols. The key idea of the CHSH based DIQKD protocol is, if the devices are honest and free from external adversarial attacks, then we should be able to observe the clear quantum advantage limit with a limited probability of error. If the error exceeds the predefined limit, then it means that the device is not secure, and so we abort the protocol.

Let the N players involved be denoted as P^1, P^2 till P^N and the devices used by the players be denoted as D^1, D^2 till D^N respectively. The initial setup is similar to

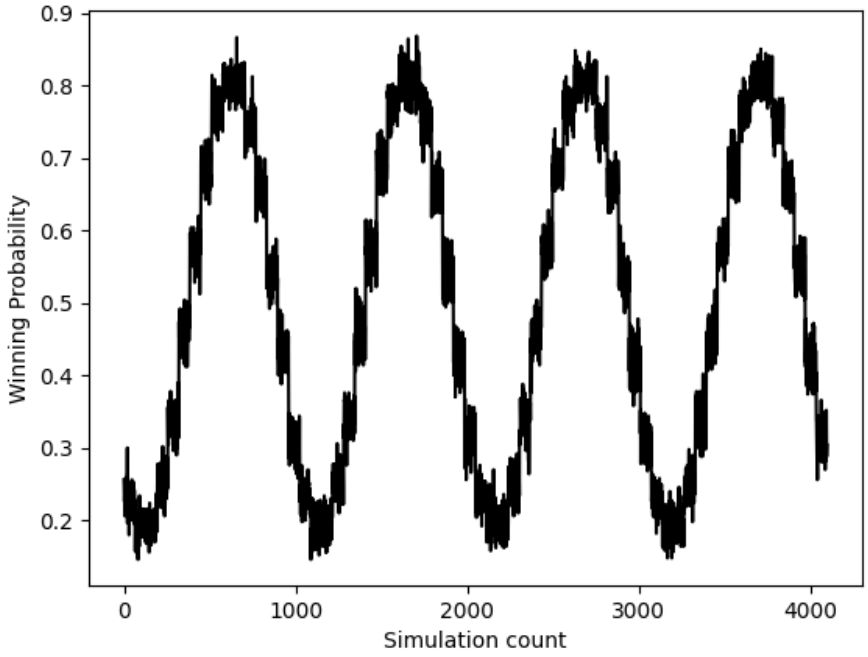

Figure 3.5: Plot of the winning probabilities for Four Qubit Game 2

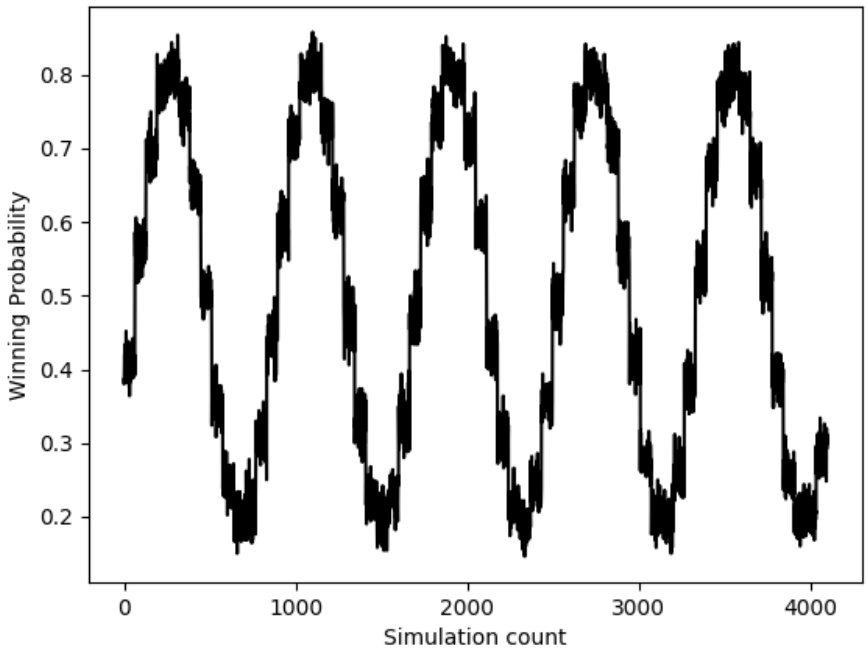

Figure 3.6: Plot of the winning probabilities for Five Qubit Game

the on explained in Section 2.1. But instead of using their own predefined strategies, here the player P^i will pass the input X^i to their device D^i and get the output A^i.

3.6 Protocol For Three Qubit System

The below algorithm explains the DIQKD protocol extended to a system of three parties.

Proposed Protocol For Three Party DIQKD

---

1. Let n be the number of input rounds and η be the noise tolerance.

2. Each of the three players holds one of the three entangled qubits from the GHZ state.

3. For rounds r = 1,2,...n: P_1 selects $X_1 \in \{0, 1, 2\}$, P_2 selects $X_2 \in \{0, 1, 2\}$ and P_3 selects $X_3 \in \{0, 1\}$. They input X_1, X_2 and X_3 into their devices D_1, D_2 and D_3, thus obtaining the outputs A_1, A_2 and A_3 respectively.

4. Testing Phase : If $X_1 \in 0, 1$, they play the three qubit CHSH game according to [10] and check whether the corresponding quantum advantage is achieved or not with a limited error (i.e) they check whether they are able to achieve the quantum advantage limit of $\cos^2(\frac{\pi}{8})$ - η. If yes, proceed to Key Generation phase, else, abort the protocol.

5. Key Generation Phase : The players publicly reveal their choices of the rounds. Let C be the set of rounds r in which $(X_{1,i}, X_{2,i}, X_{3,i}) = (2, 2, 1)$. On the additional input of $(X_1, X_2) = (2, 2)$, devices D_1 and D_2 measure in the same basis as that of D_3's on input $X_3 = 1$.

6. The players compute the fraction of rounds for which $A_1 = A_2 = A_3$. If it is less than 1 - η, they abort the protocol, else, they proceed to information reconciliation on the remaining rounds followed by privacy amplification to generate the shared key.

3.7 Protocol For Four and Five Qubit System

The protocol remains almost the same for four and five qubit system except for few changes. Instead of the three qubit CHSH game in testing phase, now we use our newly proposed four and five qubit CHSH games. For four qubit system, the devices measure in the same basis when ($X_{1,i}$, $X_{2,i}$, $X_{3,i}$, $X_{4,i}$) = (2, 2, 2, 1) and for five qubit system, the devices measure in the same basis when ($X_{1,i}$, $X_{2,i}$, $X_{3,i}$, $X_{4,i}$, $X_{5,i}$) = (2, 2, 2, 2, 1).

3.8 Implementation Results

The implementation of the proposed protocol is carried out using the IBM Qiskit python framework for Quantum Computing. A GHZ state was prepared and the winning conditions are set according to [10]. This gives a clear quantum advantage with success rate of above 85%. Then we applied the proposed algorithm to calculate the common shared key. A success rate of 80% was achieved in the common key generation which could be also further improved if other variations of the game with more better bases are used for measurements.

CHAPTER 4

PROPOSED WORK 2

In Chapter 4, we used the CHSH game based approach. Here, we will use the MPPT game based approach. But since the traditional MPPT game is for two parties and we want to design for three parties, we shall extend the traditional MPPT game to a newer one which can handle for three parties. This is essential because, even though the traditional MPPT gives a perfect success rate for n¿=3, where n denotes the no of qubits, the same approach cannot be used to derive a common shared key in the three party system. So we have to modify the whole traditional MPPT game and redefine its rules to match our three party system.

4.1 Modified MPPT Game For Three Party System

For any $n \geq 3$, game G_n consists of n players. Each player A_i receives a single input bit x_i and is requested to produce a single output bit y_i. The players are promised that the number of 1's among their inputs is a multiple of three (i.e) $\sum x_i \equiv 0 \pmod 3$. We require that,

$$\sum_{i}^{n} y_i \equiv \frac{1}{3} \sum_{i}^{n} x_i \pmod 2 \tag{4.1}$$

provided $\sum x_i \equiv 0 \pmod 3$. We say that $x = x_1x_2...x_n$ is the question and $y = y_1y_2...y_n$ is the answer.

Theorem 1. If the n players are allowed to share prior entanglement, then they can always win game G_n.

Proof. Define the following n-qubit entangled quantum states $|\phi_n^+\rangle$ and $|\phi_n^-\rangle$.

$$|\phi_n^+\rangle = \frac{1}{\sqrt{2}}(|0^n\rangle + |1^n\rangle)$$
$$|\phi_n^-\rangle = \frac{1}{\sqrt{2}}(|0^n\rangle - |1^n\rangle)$$

Let H denote the Hadamard transformation.

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$
$$H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

Let P(φ) denote the Phase Shift transformation which represents a rotation about the z-axis on the Bloch sphere by φ radians.

$$P|0\rangle = |0\rangle$$
$$P|1\rangle = e^{i\varphi}\ |1\rangle$$

The phase shift gate is represented by the matrix:

$$P(\varphi) = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\varphi} \end{bmatrix}$$

Thus we have P gate = S gate when $\varphi = \frac{\pi}{2}$. For our three party DIQKD scenario, take $\varphi = \frac{\pi}{3}$. We can see that if P($\frac{\pi}{3}$) is applied to any three qubits of $|\phi_n{}^+\rangle$, then the resulting state is $|\phi_n{}^-\rangle$. Also we can see the same in reverse that if P($\frac{\pi}{3}$) is applied to any three qubits of $|\phi_n{}^-\rangle$, then the resulting state is $|\phi_n{}^+\rangle$.

From the above calculations we can derive that if the qubits of $|\phi_n{}^+\rangle$ are distributed among the n players, and if exactly m of them apply P($\frac{\pi}{3}$) to their qubit, the resulting global state will be $|\phi_n{}^+\rangle$ if $m \equiv 0$ (mod 6) and $|\phi_n{}^-\rangle$ if $m \equiv 0$ (mod 3).

The procedure for the quantum winning strategy is given below:

1. Using the n available qubits, the state $|\phi_n{}^+\rangle$ is generated
2. Each of the n resultant qubits are distributed among the n players such that each player gets exactly one qubit
3. Each player A_i recieves an input bit x_i from the Referee R
4. If $x_i = 0$, then player A_i will not apply any transformation on his/her qubit
5. If $x_i = 1$, then player A_i will apply the P($\frac{\pi}{3}$) transformation on his/her qubit
6. Player will then apply H to his/her qubit
7. Then the player measures his/her qubit and gets the output y_i which is sent back to the referee

The referee will create the input string $x = x_1x_2...x_n$ in such a way that it always satisfies the condition $\sum x_i \equiv 0$ (mod 3). So the referee has only two options to generate the input string, either $\frac{1}{3}\sum_i^n x_i$ (mod 2) $\equiv 0$ or $\frac{1}{3}\sum_i^n x_i$ (mod 2) $\equiv 1$.

After applying the quantum strategy, we can see that $\sum_i^n y_i$ is even if $\frac{1}{3}\sum_i^n x_i$ (mod 2) $\equiv 0$ and odd if $\frac{1}{3}\sum_i^n x_i$ (mod 2) $\equiv 1$.

Thus, in both the cases, the Eqn.(4.1) is always satisfied.

Table 4.1: Possible Input Combinations in Modified MPPT Game for 4 qubits

x_1	x_2	x_3	x_4
0	0	0	0
1	1	1	0
1	1	0	1
1	0	1	1
0	1	1	1

4.2 Protocol Description and Key Generation

Let the three parties be denotes as Alice, Bob and Chuck and let the key generated by them be denoted as K_A, K_B and K_C respectively. The number of qubits required for our protocol is four. Among them, two are given to Alice. Bob and Chuck get one qubit each.

At the beginning of the protocol, n number of four qubit entangled states (i.e) $\frac{1}{\sqrt{2}}(\left|0^4\right\rangle + \left|1^4\right\rangle)$ such states are prepared or obtained from a third party (TP). The first two qubits are given to Alice, the third to Bob and the fourth to Chuck by the referee. Now, since the three parties have access to quantum resources, they play the gam using the winning quantum strategy and produce the final common shared key.

While Alice can calculate the shared key independently, Bob and Chuck need some additional support channel for interchanging their outputs. This support could be provided by any of the already existing two party DIQKD protocols to maintain the same level of DI security.

The detailed key generation process for all combinations of inputs and outputs is described in Table 4.2. From the table, we can clearly verify that $K_A = K_B = K_C$ which proves the correctness of the protocol.

4.3 Algorithm for key generation using the Modified MPPT game

The below algorithm explains the step-by-step procedure to derive th key using the Modified MPPT game.

Proposed Protocol For Three Party DIQKD using the Modified MPPT

1. Let n be the number of $\frac{1}{\sqrt{2}}(\left|0^4\right\rangle + \left|1^4\right\rangle)$ such states provided by a third party (TP).
2. Alice holds two of the four entangled qubits. Bob and Chuck hold one entangled qubit each.
3. For rounds r = 1,2,...n: Alice selects x_1,$x_2 \in \{0, 1\}$, Bob selects $x_3 \in \{0, 1\}$ and Chuck selects $x_4 \in \{0, 1\}$ uniformly at random.
4. They input x_1, x_2, x_3 and x_4 into their devices D_1, D_2, D_3 and D_4 respectively.
5. The referee R will generate the input string $x = x_1x_2...x_n$ according to the game condition and then send it to the parties.
6. The parties will perform the quantum winning strategy condition rules for the game and obtain the outputs y_1, y_2, y_3 and y_4 respectively.
7. For Alice, the key generation strategy is : $K_A = (x_1 \vee x_2) \oplus y_1 \oplus y_2$
8. For Bob, the key generation strategy is : $K_B = y_3 \oplus y_4$
9. For Chuck, the key generation strategy is : $K_B = y_3 \oplus y_4$

Table 4.2: Strategy for key generation using the Modified MPPT for Three parties

x_1x_2	x_3	x_4	y_1y_2	y_3	y_4	$K_A = (x_1 \vee x_2) \oplus y_1 \oplus y_2$	$K_B = y_3 \oplus y_4$	$K_C = y_3 \oplus y_4$
00	0	0	00	0	0	0	0	0
			11	0	0	0	0	0
			10	1	0	1	1	1
			10	0	1	1	1	1
			01	1	0	1	1	1
			01	0	1	1	1	1
			00	1	1	0	0	0
			11	1	1	0	0	0
11	1	0	10	0	0	0	0	0
			01	0	0	0	0	0
			00	1	0	1	1	1
			00	0	1	1	1	1
			11	1	0	1	1	1
			11	0	1	1	1	1
			10	1	1	0	0	0
			01	1	1	0	0	0
11	0	1	10	0	0	0	0	0
			01	0	0	0	0	0
			00	1	0	1	1	1
			00	0	1	1	1	1
			11	1	0	1	1	1
			11	0	1	1	1	1
			10	1	1	0	0	0
			01	1	1	0	0	0
10	1	1	10	0	0	0	0	0
			01	0	0	0	0	0
			00	1	0	1	1	1
			00	0	1	1	1	1
			11	1	0	1	1	1
			11	0	1	1	1	1
			10	1	1	0	0	0
			01	1	1	0	0	0
01	1	1	10	0	0	0	0	0
			01	0	0	0	0	0
			00	1	0	1	1	1
			00	0	1	1	1	1
			11	1	0	1	1	1
			11	0	1	1	1	1
			10	1	1	0	0	0
			01	1	1	0	0	0

4.4 Implementation Results

The implementation of the proposed protocol is carried out using the IBM Qiskit python framework for Quantum Computing. A four qubit entangled state was prepared and the winning conditions are set according to the Modified MPPT game. This gives a perfect quantum advantage with success rate of 100%. Then we applied the proposed algorithm for key generation to each of the parties to calculate the common shared key. A success rate of 100% was achieved in the common key generation. This is because we have assumed that all the qubits used in our protocol are ideal.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Summary

Through this work, we have proposed two approaches for designing the three party QKD system with DI security. Also, we proposed new quantum games on four qubit and five qubit systems based on the traditional CHSH game. Further, we have applied their quantum advantage property by designing DI secure QKD for their corresponding multi-party systems. DIQKD is currently an emerging research field in cryptography and security due to its higher security than the presently operational classical key distribution algorithms and its resistance to attacks even by future quantum computers.

5.2 Future Work

In future, this research could be extended further to design DIQKD protocols for higher multi-party systems (i.e) six parties or more. Also the simulation factor s could be optimized to yield results of higher accuracy. This paper assumes the usage of ideal qubits, which would be possible in the upcoming years, but for the current Noisy intermediate-scale quantum (NISQ) era, the practical applications of these protocols may give a slight deviated value from their calculated ones due to the presence of errors and interference from the external physical world. Hence adaptability of these protocols to the present NISQ era also poses a significant challenge to be handled.

CHAPTER 6

PUBLICATION OUTCOME FROM THIS RESEARCH

6.1 Conference Publication

"New Multi-Qubit CHSH Games and their application to Device-Independent Quantum Key Distribution protocols"
by
Venugopal S and Kunwar Singh

The above mentioned submission of our research work has been accepted for presentation and will appear in the proceedings of the **17th International Conference on COMmunication Systems & NETworkS - Workshop on Quantum Technologies (COMSNETS '2025 - WQT)**.

Conference Website : https://www.comsnets.org/WQT.html#accepted_papers

APPENDIX A

CODE ATTACHMENTS

A.1 Implementation Code for the Proposed Protocol 1

```
1
2  from qiskit import QuantumCircuit
3  from qiskit_aer.primitives import Sampler
4  from numpy import pi
5  from numpy.random import randint
6
7  X1 = []
8  X2 = []
9  X3 = []
10 T = []
11 A1 = []
12 A2 = []
13 A3 = []
14
15 def chsh_game(strategy):
16     x1 = randint(0, 2)
17     x2 = randint(0, 2)
18     x3 = randint(0, 2)
19     t = randint(0, 2)
20
21     a1, a2, a3 = strategy(x1, x2, x3, t)
22
23     x1, x2, x3 = int(x1), int(x2), int(x3)
24     a1, a2, a3 = int(a1), int(a2), int(a3)
25     t = int(t)
26     nx1, nx2, nx3 = 1 if (x1==0) else 0, 1 if (x2==0) else 0, 1 if (
           x3==0) else 0
27     na1, na2, na3 = 1 if (a1==0) else 0, 1 if (a2==0) else 0, 1 if (
           a3==0) else 0
28
29     X1.append(x1)
30     X2.append(x2)
31     X3.append(x3)
32     T.append(t)
33     A1.append(a1)
34     A2.append(a2)
35     A3.append(a3)
36
37
38 def chsh_circuit(x1, x2, x3, t):
39     # # Pure States
40     N_qubits=3
41     qc = QuantumCircuit(N_qubits, N_qubits)
42     qc.h(0)
43     for i in range (0, N_qubits-1):
```

```
44            qc.cx(i, i+1)
45        qc.barrier()
46
47        # # Mixed States
48        # N_qubits=4
49        # qc = QuantumCircuit(N_qubits, N_qubits)
50        # qc.h(0)
51        # for i in range (0, N_qubits-1):
52        #     qc.cx(i, i+1)
53        # qc.barrier()
54
55        # Alice
56        # t=0 means test_round; t=1 means generation_round
57        if t == 0:
58            if x1 == 0:
59                qc.rx(pi / 2, 0)
60                qc.ry((0*pi) / 4, 0)
61                qc.rz((1*pi) / 12, 0)
62            elif x1 == 1:
63                qc.rx(pi / 2, 0)
64                qc.ry((1*pi) / 2, 0)
65                qc.rz((7*pi) / 12, 0)
66        elif t == 1:
67            qc.rx(pi / 2, 0)
68            qc.ry((0*pi) / 4, 0)
69
70        # Bob
71        if t == 0:
72            if x2 == 0:
73                qc.rx(pi / 2, 1)
74                qc.ry((0*pi) / 4, 1)
75                qc.rz((1*pi) / 12, 1)
76            elif x2 == 1:
77                qc.rx(pi / 2, 1)
78                qc.ry((1*pi) / 2, 1)
79                qc.rz((7*pi) / 12, 1)
80        elif t == 1:
81            qc.rx(pi / 2, 1)
82            qc.ry((0*pi) / 4, 1)
83
84        # Chuck
85        if x3 == 0:
86            qc.rx(pi / 2, 2)
87            qc.ry((-1*pi) / 4, 2)
88            qc.rz((1*pi) / 12, 2)
89        else:
90            qc.rx(pi / 2, 2)
91            qc.ry((1*pi) / 4, 2)
92            qc.rz((7*pi) / 12, 2)
93
94        qc.barrier()
95        qc.measure(0, 0)
96        qc.measure(1, 1)
97        qc.measure(2, 2)
98        # qc.measure(3, 3)
99
```

```
100     return qc
101
102 print("(x1, x2, x3, t) = (0, 0, 1, 1)")
103 display(chsh_circuit(0, 0, 1, 1).draw('mpl', style="iqp"))
104
105 sampler = Sampler()
106
107 def quantum_strategy(x1, x2, x3, t):
108     # 'shots=1' runs the circuit once
109     result = sampler.run(chsh_circuit(x1, x2, x3, t), shots=1).result
            ()
110     statistics = result.quasi_dists[0].binary_probabilities()
111     bits = list(statistics.keys())[0]
112     a1, a2, a3 = bits[0], bits[1], bits[2]
113     return a1, a2, a3
114
115 NUM_GAMES = 1000
116
117 for _ in range(NUM_GAMES):
118     chsh_game(quantum_strategy)
119
120
121 TOTAL_SCORE = 0
122 NO_OF_GENERATION_ROUNDS = 0
123
124 for i in range(NUM_GAMES):
125     x1 = X1[i]
126     x2 = X2[i]
127     x3 = X3[i]
128     t = T[i]
129     a1 = A1[i]
130     a2 = A2[i]
131     a3 = A3[i]
132
133     nx1 = 1 if (x1==0) else 0
134     nx2 = 1 if (x2==0) else 0
135     nx3 = 1 if (x3==0) else 0
136     na1 = 1 if (a1==0) else 0
137     na2 = 1 if (a2==0) else 0
138     na3 = 1 if (a3==0) else 0
139
140     if t == 0:
141         # if (a1 ^ a2 ^ a3) == (x1 and x2) or ((x1^x2) and x3):
142         # if (a1 ^ a2 ^ a3) == (x1 and x2) or (x1 and x3) or (x2 and
              x3):
143         if (a1 ^ a2 ^ a3) == (x1 and x2 and x3) or (nx1 and nx2 and
              nx3):
144             TOTAL_SCORE += 1
145     elif t == 1:
146         NO_OF_GENERATION_ROUNDS += 1
147
148 print("Fraction of games won:", TOTAL_SCORE / (NUM_GAMES-
    NO_OF_GENERATION_ROUNDS))
149
150
151 isKeyDifferent = 0
```

```
152
153 for i in range(NUM_GAMES):
154     x1 = [i]
155     x2 = X2[i]
156     x3 = X3[i]
157     t = T[i]
158     a1 = A1[i]
159     a2 = A2[i]
160     a3 = A3[i]
161     if t == 1 and x3 == 1:
162         if (a1!=a2) or (a2!=a3):
163             isKeyDifferent += 1
164             # break
165
166 print("isKeyDifferent:", isKeyDifferent)
167
168 print("Key Success Rate = ", (NUM_GAMES - isKeyDifferent) / NUM_GAMES
    )
```

A.2 Implementation Code for the Proposed Protocol 2

```
1
2  # Required imports
3
4  from qiskit import QuantumCircuit
5  from qiskit_aer.primitives import Sampler
6  from numpy import pi
7  from numpy.random import randint
8
9
10 def mppt_game(strategy):
11     while (True):
12         x1, x2, x3, x4 = randint(0, 2), randint(0, 2), randint(0, 2),
               randint(0, 2)
13         if (((x1 + x2 + x3 + x4) % 3) == 0):
14             break;
15
16     # Use strategy to choose the output
17     y1, y2, y3, y4 = strategy(x1, x2, x3, x4)
18
19     x1=int(x1)
20     x2=int(x2)
21     x3=int(x3)
22     x4=int(x4)
23     y1=int(y1)
24     y2=int(y2)
25     y3=int(y3)
26     y4=int(y4)
27
28     # Referee decides if the players win or lose
29     if ( ((x1 + x2 + x3 + x4)/3) != ((y1 + y2 + y3 + y4)%2)):
30         # print("FAILED")
31         return 0
32
33     # Generate the keys
34     x1_or_x2 = 0
35     if (x1==1 or x2==1):
36         x1_or_x2=1
37
38     alice_calculated = (int(x1_or_x2) + int(y1) + int(y2)) % 2
39
40     bob_calculated = (int(y3) + int(y4)) % 2
41
42     chuck_calculated = (int(y3) + int(y4)) % 2
43
44     # print(alice_calculated, bob_calculated, chuck_calculated)
45
46     if ((alice_calculated == bob_calculated) and (bob_calculated ==
           chuck_calculated)):
47         return 1  # Win
48     return 0  # Lose
49
50
51
```

```
52
53 def mppt_circuit(x1, x2, x3, x4):
54     """
55         Creates a 'QuantumCircuit' that implements the best MPPT
               strategy.
56     Args:
57         x1 (int): Alice's bit (must be 0 or 1)
58         x2 (int): Alice's bit (must be 0 or 1)
59         x3 (int): Bob's bit (must be 0 or 1)
60         x4 (int): Chuck's bit (must be 0 or 1)
61     Returns:
62         QuantumCircuit: Circuit that, when run, returns Alice, Bob
               and Chuck's answer bits.
63     """
64     N_qubits = 4
65     qc = QuantumCircuit(N_qubits)
66     qc.h(0)
67     qc.cx(0, range(1, N_qubits))
68     qc.barrier()
69
70     # Alice
71     if x1 == 1:
72        qc.rz(-pi / 3, 0)
73
74     if x2 == 1:
75         qc.rz(-pi / 3, 1)
76
77     # Bob
78     if x3 == 1:
79         qc.rz(-pi / 3, 2)
80
81     # Chuck
82     if x4 == 1:
83         qc.rz(-pi / 3, 3)
84
85     for i in range(N_qubits):
86         qc.h(i)
87
88     qc.measure_all()
89
90     return qc
91
92
93
94
95 print("(x1, x2, x3, x4) = (0,1,0,1)")
96 display(mppt_circuit(0, 1, 0, 1).draw('mpl', style="iqp"))
97
98 sampler = Sampler()
99
100 def quantum_strategy(x1, x2, x3, x4):
101     """
102         Carry out the best strategy for the MPPT game.
103     Args:
104         x1 (int): Alice's bit (must be 0 or 1)
105         x2 (int): Alice's bit (must be 0 or 1)
```

```
106             x3 (int): Bob's bit (must be 0 or 1)
107             x4 (int): Chuck's bit (must be 0 or 1)
108         Returns:
109             (int, int, int, int): Alice, Bob and Chuck's answer bits (
                    respectively)
110         """
111         # 'shots=1' runs the circuit once
112         result = sampler.run(mppt_circuit(x1, x2, x3, x4), shots=1).
                result()
113         statistics = result.quasi_dists[0].binary_probabilities()
114         bits = list(statistics.keys())[0]
115         a, b, c, d = bits[0], bits[1], bits[2], bits[3]
116         return a, b, c, d
117
118
119
120 NUM_GAMES = 1000
121 TOTAL_SCORE = 0
122
123 for _ in range(NUM_GAMES):
124     TOTAL_SCORE += mppt_game(quantum_strategy)
125
126 print("Fraction of games won:", TOTAL_SCORE / NUM_GAMES)
```

REFERENCES

[1] *John Preskill, Course Information for Physics 219 / Computer Science 219 Quantum Computation (Formerly Physics 229)*. URL: http://theory.caltech.edu/~preskill/ph229/.

[2] *Ryan O'Donnell and John Wright, 15-859BB: Quantum Computation and Information 2015*. URL: http://www.cs.cmu.edu/~odonnell/quantum15/.

[3] Charles Bennett and Gilles Brassard. "WITHDRAWN: Quantum cryptography: Public key distribution and coin tossing". In: *Theoretical Computer Science - TCS* 560 (Jan. 1984), pp. 175–179. DOI: 10.1016/j.tcs.2011.08.039.

[4] Peter W. Shor and John Preskill. "Simple Proof of Security of the BB84 Quantum Key Distribution Protocol". In: *Phys. Rev. Lett.* 85 (2 2000), pp. 441–444. DOI: 10.1103/PhysRevLett.85.441. URL: https://link.aps.org/doi/10.1103/PhysRevLett.85.441.

[5] Artur K. Ekert. "Quantum cryptography based on Bell's theorem". In: *Physical Review Letters* 67 (6 1991), pp. 661–663. DOI: 10.1103/PhysRevLett.67.661. URL: https://link.aps.org/doi/10.1103/PhysRevLett.67.661.

[6] A. Einstein, B. Podolsky, and N. Rosen. "Can Quantum-Mechanical Description of Physical Reality Be Considered Complete?" In: *Phys. Rev.* 47 (10 1935), pp. 777–780. DOI: 10.1103/PhysRev.47.777. URL: https://link.aps.org/doi/10.1103/PhysRev.47.777.

[7] John Clauser, Michael Horne, Abner Shimony, and R. Holt. "Proposed Experiment to Test Local Hidden-Variable Theories". In: *Physical Review Letters* 23 (Oct. 1969), pp. 880–884. DOI: 10.1103/PhysRevLett.23.880.

[8] Umesh Vazirani and Thomas Vidick. "Fully Device-Independent Quantum Key Distribution". In: *Physical Review Letters* 113 (14 2014), p. 140501. DOI: 10.1103/PhysRevLett.113.140501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.113.140501.

[9] Jyotirmoy Basak, Arpita Maitra, and Subhamoy Maitra. "Device Independent Quantum Key Distribution Using Three-Party Pseudo-Telepathy". In: *Progress in Cryptology – INDOCRYPT 2019*. Ed. by Feng Hao, Sushmita Ruj, and Sourav Sen Gupta. Cham: Springer International Publishing, 2019, pp. 456–

471. ISBN: 978-3-030-35423-7. DOI: 10.1007/978-3-030-35423-7_23.

[10] Hamza Jaffali and Frédéric Holweck. "Two new non-equivalent three-qubit CHSH games". In: *Quantum Information and Computation* 24.5-6 (2024), pp. 0438–0454. DOI: 10.26421/QIC24.5-6-4. arXiv: 2312.00578 [quant-ph].

[11] J. S. Bell. "On the Einstein Podolsky Rosen paradox". In: *Physics Physique Fizika* 1 (3 1964), pp. 195–200. DOI: 10.1103/PhysicsPhysiqueFizika.1.195.

[12] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.

[13] Gilles Brassard, Anne Broadbent, and Alain Tapp. "Multi-Party Pseudo-Telepathy". In: *Lecture Notes in Computer Science* 2748 (July 2003). DOI: 10.1007/978-3-540-45078-8_1.