JOURNAL OF MECHANICS OF CONTINUA AND MATHEMATICAL SCIENCES
www.journalimcms.org

ISSN (Online) : 2454 -7190 Vol.-20, No.-8, August (2025) pp 101 - 111 ISSN (Print) 0973-8975

QUANTUM KEY DISTRIBUTION USING SUPER DENSE CODING

Tamal Deb[1], Jyotsna Kumar Mandal[2], Deeptanu Sen[3]

[1,3] Computer Sc. Department., Barasat College, Kolkata, India.

[2]Computer Sc. & Engg. Department., University of Kalyani, Nadia, India.

Email: [1]tamaldeb.kol65@gmail.com, [2]jkm.cse@gmail.com
[3]deeptanusends2005@gmail.com

Corresponding Author: **Deeptanu Sen**

https://doi.org/10.26782/jmcms.2025.08.00007

(Received: May 23, 2025; Revised: July 26, 2025; Accepted: August 09, 2025)

Abstract

Built based on the fundamental principles of quantum mechanics, Quantum Key Distribution (QKD) enables secure communication for distant parties. Entanglement-based protocols are a type of QKD protocol that uses the phenomenon of entanglement for detecting eavesdroppers between two communicating parties. In this paper, a novel QKD protocol is devised that uses the concept of superdense coding and padding bits to share the one-time pad, i.e., the key. The super dense coding is achieved by sharing a pre-existing entangled pair of qubits by leveraging the beautiful property of entanglement. The communicating parties can share a one-time pad using this protocol securely. This paper will demonstrate this phenomenon using the proposed protocol by showing the experimental results which has been surfaced with IBM Qiskit simulator, and the simulation establishes the applicability of the protocol and shows its effectiveness in detecting eavesdropping attempts while being simple to implement.

Keywords: Entanglement, Guard Qubit, QKD, Qiskit, Secret Key.

I. Introduction

In today's interconnected world with rising cyber-threats and the looming potential for quantum-enabled attacks, as explained by Pirandola et al. [VI], the quantum key distribution (QKD) has emerged as a groundbreaking solution. Unlike traditional cryptographic algorithms, which rely on mathematical models and computational assumptions and are vulnerable to the advances in quantum computing, Gao et al. [IV] has shown that the QKD leverages the principles of quantum mechanics to establish secure key agreements resistant to both quantum and classical adversaries in long range communication also and Cariolaro [III] has shown this in his work. QKD exploits the inherent randomness of quantum states to generate a secret key through a

quantum channel. This process is further strengthened by the no-cloning theorem, which has already proven that a quantum state cannot be cloned, ensuring the uniqueness of the generated key. Additionally, Werner Heisenberg's Uncertainty Principle reveals the presence of an eavesdropper attempting to intercept and retransmit quantum states. Any measurement attempt on the quantum system inevitably alters it, thus alerting the communicating parties to the presence of an intruder, as shown by Mina and Simion [VIII].

In the case of Quantum Computing (QC), it processes the quantum information in the form of a quantum bit(qubit), as discussed by Gujar [X], which is the physical carrier of the quantum information. A qubit can be thought of as the quantum version of the classical bit, and to describe the quantum state of a qubit, the symbols $|0\rangle$ and $|1\rangle$ are used. The basic but major difference between a classical bit and a quantum bit is: the classical bit can represent only one of two states, either *0* or *1*, whereas the quantum bit can be in any superposition of $|0\rangle$ and $|1\rangle$. Hence, a 2D-column vector of real or complex numbers, whose norm is *1*, such vector can represent the quantum state of a qubit. Mathematically, for any two complex numbers α and β, the vector $\begin{bmatrix}\alpha\\ \beta\end{bmatrix}$ can represent the state of a qubit where $|\alpha|^2 + |\beta|^2 = 1$. Another important property of quantum theory is entanglement. Mermin [V] has shown that quantum entanglement produces such a state where two systems are so strongly correlated that knowing information about a system immediately generates knowledge about the other system, even though the systems may be located far apart in the Universe, and this is the key concept of quantum teleportation.

II. The Proposed Protocol

The authors have devised a QKD protocol that uses a circuit of Superdense coding as depicted in *Figure 1,* and the protocol prepends and appends padding bits of the same length to the one-time pad, and the length of the padding is announced by the sender(Alice) in the classical post-processing phase. Since the proposed protocol uses both the classical post-processing and super dense circuit, the protocol is a hybrid of *"Prepare and Measure"*(PM) and *"Entanglement Based"*(EB) class, as discussed by Nielsen and Chuang [XII]. Here in this protocol, a 2-qubit quantum channel is used and a pair of qubits ∈ *{00,01,10,11}* is processed and transmitted simultaneously as key, i.e., one-time pad. A set of padding bits ∈ *{10,11}* is used. A padding bit stream of length *p* is appended as well as prepended to the one-time pad of length *N,* where *p* and *N* both are even numbers, because the bits, for both the one-time pad and the padding, are created in pairs. As a consequence, a total of *"p+N+p"* bits are needed to transmit to exchange a one-time pad of length N between the communicating parties.

III. Selection of Padding Bits

While searching for the best pattern for padding bits, the aim was to find such pattern that will generate the maximum number of bit mismatches between the sender(Alice) and receiver(Bob) in the presence of the Eavesdropper(Eve). For the experiment, a padding bits stream of size $p = 400$ was considered. This bit stream was transmitted through the circuit for *100* iterations following *Algorithm 1*. There was no mismatch in sent and received padding bits in the absence of Eve. But in the presence

of Eve, the protocol has found the percentage of average bit mismatch between Alice and Bob as per Table 1.

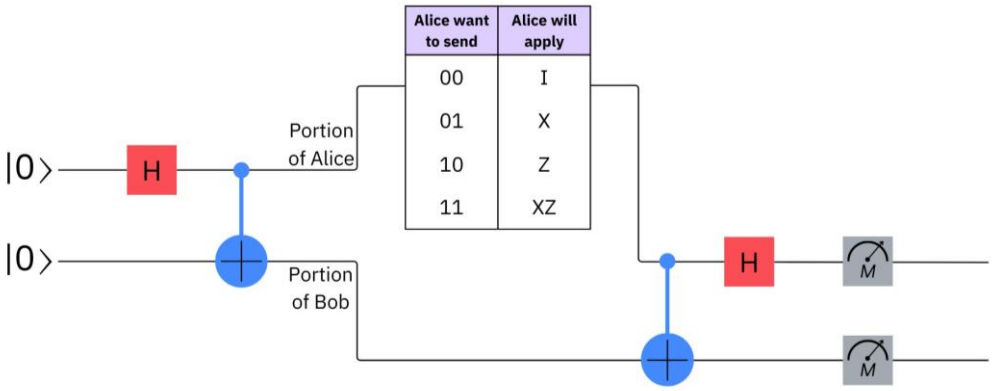

Fig. 1. The circuit used for communication

Table 1: Mismatch in Padding Bits for Different Padding Patterns

Percentage	Bit-pair Patterns of Pre and Post Padding of Length p Each
24.89% ≈ 25%	Padding created with *p/2* number of *00*
24.90% ≈ 25%	Padding created with *p/2* number of *01*
49.94% ≈ 50%	Padding created with *p* number of random *{0,1}*
75.06% ≈ 75%	Padding created with *p/2* number of *10*
74.85% ≈ 75%	Padding created with *p/2* number of *11*
75.05% ≈ 75%	Padding created with *p/2* number of *{10,11}*

The experimental facts are plotted in *Figure 2* for better visualization. Guided by the results of the experiments, the proposed protocol has created padding bits with randomly generated *10* or *11* from the set *{10,11}*. Such *p/2* elements from the said set of *{10,11}* are taken to form a pre- and post-padding of length *p* each. After the creation of padding bits, the proposed protocol moves to the next phase, i.e. *Encoding Phase*.

IV. Encoding of Padding at Sender End

During Encoding phase, *N* number of random classical bits ∈ *{0,1}*, which will be the actual one-time pad(i.e. the key), are generated and *p* number of padding bits are appended and prepended to the one-time pad and this total of *p+N+p* number of classical bits are converted to the corresponding qubits ∈ *{|0⟩, |1⟩}*. From this stream of qubits, pair q_iq_{i+1} is selected where *i = 0, 2, 4, ... ,(p + N + p)- 2.* For each pair of qubits q_iq_{i+1}, quantum gates are applied on q_i as per the following logic described in *Algorithm 1*. The whole process of encoding is executed at the end of Alice. Alice also stores the

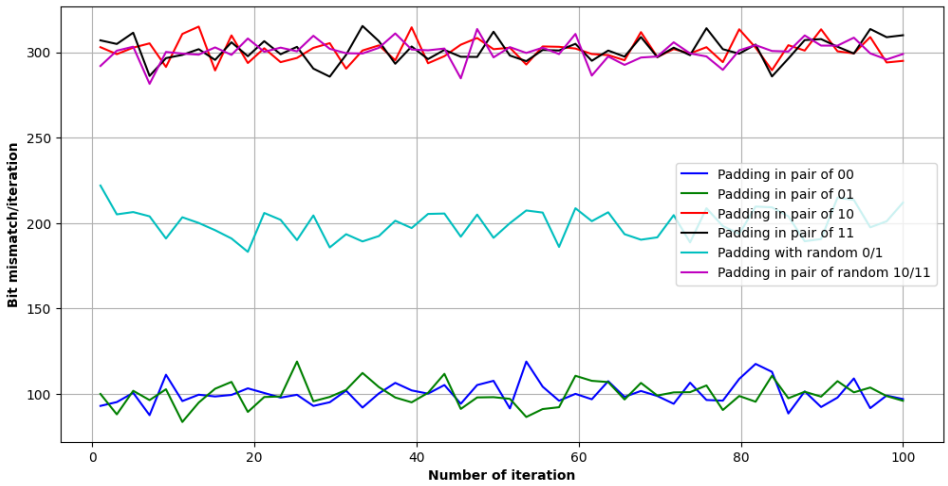

Fig. 2. Mismatch in Padding Bits for Different Padding Patterns

one-time pad and the padding bit stream for future. A test of intrusion detection will be executed after Bob receives the communicated bits.

V. Decoding at Receiver End

In the Decoding phase, Bob applies a quantum *Controlled-Not(CX)*-gate on pair q_iq_{i+1} with qi as control bit and q_{i+1} as target bit. After that, a quantum *Hadamard*-gate is applied to q_i. Finally, Bob measures the pair q_iq_{i+1}. The decoding is done by following the logic described in *Algorithm 2*. During the communication, if Eve(the intruder) is present, she also applies the same logic of *Algorithm 2,* and after each measurement, she transmits her results (i.e., pair q_iq_{i+1}) through the quantum channel without interrupting the communication so that no one can trace her.

VI. Intrusion Detection Phase

On the completion of the communication, over a classical channel, sender Alice announces one decimal number *p*, which is the length of each padding. From the measurement outcome, Bob extracts the prepended and appended *p* and *p* classical bits and makes a classical *XOR* of them. If the result turns out to be *0*, it means a successful transmission without any intrusion; otherwise, the result is discarded and the whole process is started again. The logic of intrusion detection is described in *Algorithm 3*.

Algorithm 1: Encoding at Sender End

1: TotalKeySize ← $p + N + p$
2: AliceSent ← []
3: AliceSent ← AliceSent ∪ *p padding bits*
4: AliceSent ← AliceSent ∪ *N randomly generated Key bits*
5: AliceSent ← AliceSent ∪ *p padding bits*
6: **for** $i = 0, 2, 4, \ldots, TotalKeySize - 2$ **do**
7: Create a Quantum circuit with 2-qubits q_0 *and* q_1
8: *Hadamard* gate is applied on q_0

Controlled-Not gate is applied on q_1 with q_0 as control bit
if *AliceSent*[*i*] = *0* **and** *AliceSent*[*i* + *1*] = *0* **then**
Apply *Identity* gate on q_0
else
if *AliceSent*[*i*] = *0* **and** *AliceSent*[*i* + *1*] = *1* **then**
Apply *Pauli-X* gate on q_0
else
if *AliceSent*[*i*] = *1* **and** *AliceSent*[*i* + *1*] = *0* **then**
Apply *Pauli-Z* gate on q_0
else
Apply *Pauli-X* gate on q_0
Apply *Pauli-Z* gate on q_0
end if
end if
end if
end for

Algorithm 2: Decoding at Receiver End

BobFound ← []
for *i = 0, 2, 4, ..., TotalKeySize − 2* **do**
Controlled-Not gate is applied on q_1 with q_0 as control bit
Hadamard gate is applied on q_0
Classical bit c_0 ← measurement of q_0
Classical bit c_1 ← measurement of q_1
BobFound[i] ← c_0 and BobFound[i+1] ← c_1
end for

Algorithm 3: Detection of Intrusion

Alice announces decimal number *p* over classical channel
Bob sets PrePadding ← []
Bob sets PostPadding ← []
for *i = 0, 1, 2, ..., p-1* **do**
PrePadding[i] ← BobFound[i]
PostPadding[i] ← BobFound[i+p+N]
end for
if PrePadding[*0, 1, ..., p-1*] *Classical-XOR* PostPadding[*0 ,1, ..., p-1*] = 0 **then**
No Intrusion, Return BobFound[*p, p+1, ..., p+N-1*] as *one-time pad*
else
Intrusion detected, Return *FALSE*
end if

VII. Results and Comparisons

Proof of no increment or no injection of entanglement

Here, in this work, there are two trusted parties, Alice(A) and Bob(B), and the third party is the eavesdropper Eve(C). The parties A and B create any of the following Bell states $|\phi^+\rangle$, $|\phi^-\rangle$, $|\Psi^+\rangle$ and $|\Psi^-\rangle$ where:

$$|\phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle),$$
$$|\phi^-\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle),$$
$$|\Psi^+\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle) \text{ and}$$
$$|\Psi^-\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle).$$

Inherently, these Bell States are maximally entangled. Therefore, A is maximally entangled with B, meaning entanglement, of A and B is maximal, which is shown in Equation (i) as follows:

$$E_{AB} = 1 \tag{1}$$

If possible, the third-party C tries to inject entanglement locally. But the monogamous property of entanglement states that entanglement cannot be injected between two systems using only local operations and classical communication. Equation (ii) states the monogamy inequality for qubits:

$$E_{A|BC} \geq E_{AB} + E_{AC} \tag{2}$$

where E_{XY} is an entanglement measure between the two systems of qubits X and Y.

For any maximally entangled Bell state between A and B, for example, $|\phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$, it is assumed that there exists a third system C in a state $|\Psi\rangle_C$, and tries to entangle it with either A or B via a unitary $|U\rangle_{AC}$, therefore, the new system will be in state, shown in Equation (iii) as follows:

$$|\Psi\rangle_{ABC} = U_{AC}(|\phi^+\rangle_{AB} \otimes |\Psi\rangle_C) \tag{3}$$

Since unitary operations preserve entanglement, and the third, system is acting locally, the total entanglement between A and B cannot be increased.

Therefore, as per Equation (i) and the monogamy inequality (ii), $E_{AC} = 0$ and it holds always for any local operation.

Hence, it is proved that, in this work, no third party can increase or inject entanglement during communication. Because communication in this work is performed based on Bell states $|\phi^+\rangle$, $|\phi^-\rangle$, $|\Psi^+\rangle$ and $|\Psi^-\rangle$ as shown in the used circuit in *Figure 1*.

Analytical Framework parameters

As demonstrated by Ekert [I] and discussed by Portmann and Renner [XI], the trace distance is a measure of the distinguishability between the actual state of the key known to Eve after the key transmission and the ideal state of the key. If the final key is ρ_{ke} and the ideal state is $\tau_k \otimes \rho_e$ and if the key is ϵ-secure, then:

$$D = \frac{1}{2} || \rho_{ke} - \tau_k \otimes \rho_e ||_1 \leq \epsilon \quad (4)$$

where τ_k = uniform random key, ρ_e = Eve's marginal state and ϵ is the bound on the information gathered by Eve about the key.

The proposed protocol is based on entangled qubits, and to check the presence of Eve, the CHSH inequality is used here. The analytical framework includes:

- Transmission of the entangled padding and key bits. Total Transmission TT: $p + N + p$
- Key shifting, i.e., the number of bits that can be used as key: N, and the $p + p$ padding bits are tested and discarded, hence, shifted key size = N
- Estimation of Quantum Bit Error Rate(QBER)
- Computation of CHSH parameter, S, for security test
- Leak to Eve during error correction:
  $Leak = f * N * h(QBER)$, where f is the error correction inefficiency factor
- Estimation of the lower bound of Eve's knowledge(smooth min-entropy):
  $$H_{min}^{\epsilon}(X|E) \geq N * (1 - h(QBER)$$
- Final key length:
  $L \leq H_{min}^{\epsilon}(X|E) - Leak - 2\log_2\left(\frac{1}{\epsilon_{PA}}\right)$, where ϵ_{PA} is privacy amplification error

In *Table 2*, column T represents the total padding used in the proposed work, where $T = total\ of\ pre\ and\ post\ padding = p + p$, which is shown in different percentage values of total transmitted entangled pairs(*TT*), i.e., *T* is *10% of TT*, or *20% of TT,* and so on. *Table 2* shows the mentioned parameters of the proposed protocol, and *Table 3* shows the same for the E91 protocol for $\epsilon_{PA} = 10^{-6}$ and $f = 1.2$.

Table 2: Leak, Smooth min-entropy and Final Key length of Proposed Work

TT	T	N=TT-T	$QBER$	S	f	$h(QBER)$	$Leak$	$H_{min}^{\epsilon}(X\|E)$	L
10^5	10%	90000	2%	2.6	1.2	≈ 0.141441	≈15276	≥ 77270	≤ 61954
10^5	10%	90000	3%	2.6	1.2	≈ 0.194392	≈20994	≥ 72505	≤ 51471
10^5	10%	90000	4%	2.6	1.2	≈ 0.242292	≈26167	≥ 68194	≤ 41987
10^5	20%	80000	2%	2.6	1.2	≈ 0.141441	≈13578	≥ 68685	≤ 55067
10^5	20%	80000	3%	2.6	1.2	≈ 0.194392	≈18661	≥ 64449	≤ 45748
10^5	20%	80000	4%	2.6	1.2	≈ 0.242292	≈23260	≥ 60617	≤ 37317
10^5	30%	70000	2%	2.6	1.2	≈ 0.141441	≈11881	≥ 60099	≤ 48178
10^5	30%	70000	3%	2.6	1.2	≈ 0.194392	≈16329	≥ 56392	≤ 40023
10^5	30%	70000	4%	2.6	1.2	≈ 0.242292	≈20352	≥ 53039	≤ 32647
10^5	50%	50000	2%	2.6	1.2	≈ 0.141441	≈8486	≥ 42928	≤ 34402
10^5	50%	50000	3%	2.6	1.2	≈ 0.194392	≈11663	≥ 40280	≤ 28577
10^5	50%	50000	4%	2.6	1.2	≈ 0.242292	≈14537	≥ 37885	≤ 23308

Table 3: Leak, Smooth min-entropy and Final Key length of E91

TT	$N = TT/4$	$QBER$	S	f	$h(QBER)$	$Leak$	$H_{min}^{\epsilon}(X\|E)$	L
10^5	25000	2%	2.6	1.2	≈ 0.141441	≈4243	≥ 21464	≤ 17181
10^5	25000	3%	2.6	1.2	≈ 0.194392	≈5832	≥ 20140	≤ 14268
10^5	25000	4%	2.6	1.2	≈ 0.242292	≈7269	≥ 18943	≤ 11634

Key Rate per Transmitted Qubit

The asymptotic key rate per entangled pair that corresponds to one transmitted qubit per party can be approximated by Equation (v) as:

$$R = p_{shift}(\,1 - h(QBER) - \,f * h(QBER)) \tag{5}$$

where $h(QBER) = \,-QBER * \log_2 QBER - (1 - QBER) * \log_2(1 - QBER)$, and f is the error correction inefficiency and $p_{shift} = Shifting\ Probability\ of\ Key$.

In this work, $p_{shift} = \frac{Numner\ of\ Bits\ can\ be\ used\ as\ Key}{Total\ Transmitted\ Bits} = \frac{N}{p+N+p}$, because of the pre and post padding of length p each, and with N as the precise length of the key. Hence, the $total\ transmitted\ bits = p + N + p$. Since padding length can vary in size as $p = x\%\ of\ N$, *Table 4* shows the measurements of R with different $QBER$, f, and p values.

Table 4: Asymptotic Key Rate per Entangled Pair in Proposed Protocol

$QBER$	f	p	p_{shift}	$h(QBER)$	R
2%	1.2	10% of N	5/6	≈ 0.141441	≈ 0.574023
3%	1.2	10% of N	5/6	≈ 0.194392	≈ 0.476948
4%	1.2	10% of N	5/6	≈ 0.242292	≈ 0.389131
2%	1.2	20% of N	5/7	≈ 0.141441	≈ 0.492021
3%	1.2	20% of N	5/7	≈ 0.194392	≈ 0.408812
4%	1.2	20% of N	5/7	≈ 0.242292	≈ 0.333541

The following *Table 5* shows the measurements of R for the E91 protocol with different $QBER$ and f values, and p_{shift} is taken as $1/2$.

Table 5: Asymptotic Key Rate per Entangled Pair in E91 Protocol

$QBER$	f	p_{shift}	$h(QBER)$	R
2%	1.2	1/2	≈ 0.141441	≈ 0.344415
3%	1.2	1/2	≈ 0.194392	≈ 0.286169
4%	1.2	1/2	≈ 0.242292	≈ 0.233479

Bit Mismatch in Simulation with Qiskit

According to *Algorithms 1, 2, and 3* described above, and using the IBM Qiskit, as demonstrated by Shaik and Nakkeeran [VII] in their work, quantum circuits are prepared and executed for *400* padding qubits. The circuit system is executed for *100* iterations. For each iteration, on average, the proposed protocol has found *300* mismatched bits in padding in the presence of Eve, and no mismatch is found in the

absence of Eve. As explained by Reddy et al. [IX] and demonstrated by Bennett [II], the benchmark BB84 protocol, in the presence of Eve, has traced a positional bit mismatch in *150* bits on average out of *400* bits. *Figure 3* shows a comparison of bit mismatch between the proposed protocol and BB84. Mathematically, the eavesdropper detection capability of the proposed protocol can be found as follows:

$$\frac{300*100}{400} = 75\% \tag{9}$$

Similarly, from the experimental results stated above, the intrusion detection ability of the BB84 protocol is calculated as:

$$\frac{150*100}{400} = 37.5\% \tag{10}$$

Table 5 shows the feature-wise comparison of the proposed protocol and the benchmark BB84 protocol.

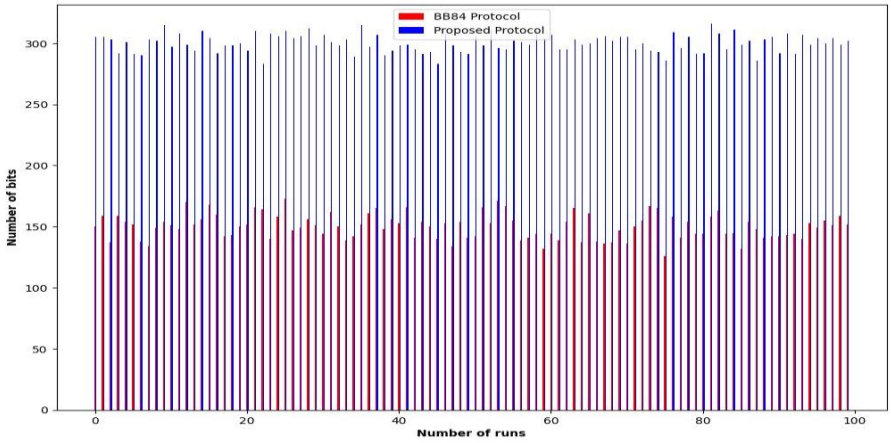

Fig. 3. Bit Mismatch: Proposed Protocol vs. BB84 Protocol Table

Table 5: Core Comparison between Proposed Protocol and BB84

Feature	Proposed Protocol	BB84 Protocol
Basis Announcement	Not Needed	Both Alice and Bob
Time Efficiency	Alicebannouncesa single decimal number	Two-wayclassical communication
Effect of intrusion	300 out of 400 bit mismatches, i.e., 75%	150 out of 400 bit mismatches, i.e., 37.5%
Eve Detection Accuracy	Much higher in terms of bit mismatch	Lower

VIII. Conclusions

The experimental facts have surfaced that the overhead of the classical post-processing phase can be decreased as minimum as the announcement overhead of a single and small decimal number. No generation of a random basis at both random of communication also hugely reduces the overall running time of the QKD process. As

a whole, the proposed protocol can detect any intrusion much faster and with lesser computational overhead.

IX. Acknowledgment

The authors want to acknowledge the Department of Computer Science & Engineering, University of Kalyani, India, for the fulfillment of this proposed research. The authors also acknowledge the support provided by the DST PURSE Scheme, Government of India, at the University of Kalyani.

Conflict of Interest:

There was no relevant conflict of interest regarding this paper.

References

I. Bennett, C. H. and Brassard, G. "Quantum cryptography: Public key distribution and coin tossing.", International Conference on Computers, Systems and Signal Processing, India, pp. 175-179, (1984). 10.1016/j.tcs.2014.05.025.

II. Cariolaro, G. "Quantum communications". Springer Vol. 2, (2015), 10.1007/978-3-319-15600-2

III. Ekert, A. K. "Quantum cryptography based on Bell's Theorem". Physical Review Letter, Vol. 67, Issue 6, pp. 661-663, (1991), 10.1103/PhysRevLett.67.661

IV. Gao, F., Liu, B., Wen, Q., Chen, H. "Quantum Key Distribution: Simulation and Characterizations". Elsevier Procedia Computer Science, Volume 65, pp. 701, (2015), 10.1016/j.procs.2015.09.014

V. Gujar S.S. "Exploring Quantum Key Distribution". 2nd DMIHER Int. Conf. on Artificial Intelligence in Healthcare, Education and Industry (IDICAIEI), pp. 1–6. IEEE, (2024), 10.1109/IDICAIEI61867.2024.10842847

VI. Mermin, N. D. "Quantum Computer Science: An Introduction." Cambridge University Press, ISBN-13: 978-0521876582. (2007)

VII. Mina, M.Z., Simion, E. "A Scalable Simulation of the BB84 Protocol Involving Eavesdropping". Innovative Security Solutions for Information Technology and Communications, pp. 91–109, Springer International Publishing, Cham, (2021), 10.1007/978-3-030-69255-1_7

VIII. Nielsen, M. A. and Chuang, I. L. "Quantum Computation and Quantum Information", Cambridge University Press, ISBN-13: 978-0521635035, (2000).

IX. Pirandola, S. et al. “Advances in quantum cryptography”. Adv. Opt. Photonics 12, pp. 1012–1236, (2020), 10.1364/AOP.361502

X. Portmann, C. and Renner, R. "Cryptographic security of quantum key distribution". arXiv:1409.3525v1, (2014), 10.48550/arXiv.1409.3525

XI. Reddy, S., Mandal, S. and Mohan, C. “Comprehensive Study of BB84, A Quantum Key Distribution Protocol, (2023), 10.13140/RG.2.2.31905.28008.

XII. Shaik E. H. and Nakkeeran R. “Implementation of Quantum Gates based Logic Circuits using IBM Qiskit”. International Conference on Computing, Communication & Security, (2020), 10.1109/ICCCS49678.2020.9277010