Tree Quantum Key Agreement Protocol for Secure Multiparty Communication

Rima Djellab[1], Youssouf Achouri[2], Malak Emziane[3], and Lyamine Guezouli[3]

[1]*LAMIE Laboratory, University of Batna 2, Batna, Algeria,*
[2]*LASTIC Laboratory, University of Batna 2, Batna, Algeria,*
[3]*LEREESI Laboratory, HNS-RE2SD, Batna, Algeria*

https://doi.org/10.26636/jtit.2024.4.1711

Abstract — The paper introduces a tree multiparty quantum key agreement protocol for secure communication between multiple participants, specifically tailored for tree topologies. Based on the BB84 protocol, the proposed solution employs hierarchical tree structures and XOR operations to facilitate efficient and secure key generation. Key elements are exchanged among participants in an equitable manner, ensuring that each participant contributes equally to the generation of the shared key. The protocol demonstrates robust security, effectively defending against both external and internal attacks, and achieves a quantum efficiency of ½(*N*–1), where *N* is the number of participants. Additionally, the protocol is readily implementable with current quantum technologies, utilizing single-photon transmission to facilitate secure key distribution.

Keywords — BB84 protocol, quantum efficiency, quantum key agreement, tree topology

1. Introduction

Cryptography protocols are essential for establishing secure communication channels. Key distribution is a vital process for creating these mechanisms, as it enables the exchange of secret data between remote parties over an insecure channel. Although most key distribution protocols utilize asymmetric algorithms [1], their security is tied to the difficulty of solving specific mathematical problems, ensuring computational security within traditional computing environments. Nevertheless, the fact that computational capabilities of potential eavesdroppers increase due to technological progress could render these protocols ineffective.

Asymmetric algorithms, such as RSA or ECC [2] rely on the mathematical difficulty of such problems as factorizing large prime numbers or solving discrete logarithms. The security of these protocols is predicated on computational infeasibility, meaning they are secure as long as solving the underlying mathematical problem remains computationally too time-consuming.

However, with the advent of quantum computing, the efficacy of these traditional protocols tends to be insufficient, as quantum computers – leveraging principles of quantum mechanics – promise to solve these complex problems in polynomial time.

BB84 [3] is a protocol that is commonly used in the field of quantum key distribution, as it facilitates the generation of a completely secure key at the end of exchanges. This key may be subsequently employed in a one-time pad (OTP) protocol to secure the communication channels.

One significant advancement in this field is the multiparty quantum key agreement (MQKA) protocol which extends the benefits of quantum key distribution to multiple parties. MQKA allows a group of participants to establish a shared secret key, ensuring that all parties are equally involved in the key generation process [4]. This is particularly useful in scenarios where collaboration among multiple entities is required. By leveraging entanglement and quantum communication channels, MQKA protocols provide robust security guarantees and are resilient to quantum attacks, ensuring integrity and confidentiality of data.

With that borne in mind, we propose a tree MQKA (T-MQKA) protocol tailored to facilitate key agreement within a tree-organized group. By leveraging the proven security of quantum principles, T-MQKA aims to fortify communication channels, ensuring their resilience against emerging threats posed by quantum computing advancements. The rest of this paper is organized as follows. A brief overview of classic and quantum key management approaches is presented in Section 2, with preliminaries and related work following in Sections 3 and 4, respectively. In Section 5, the proposed protocol is presented. A comparative analysis of our T-MQKA protocol benchmarking it against solutions belonging to a similar category (tree topology), with an emphasis placed on their quantum efficiency, is given in Sections 6–7. In Section 8, a security analysis is provided and, finally, conclusions are drawn in Section 9.

2. Classic Versus Quantum Key Management

In classic contexts, key management protocols fall into three main categories: centralized, decentralized, and distributed [5]. In centralized protocols, a central entity oversees key management and distribution within the group. Decentralized protocols distribute this responsibility among multiple entities, mitigating issues such as bottlenecks. This catego-

ry further distinguishes between time-oriented decentralized protocols, where key changes occur at set intervals, and dynamic-oriented protocols, where key adjustments accommodate member departures or arrivals to maintain confidentiality.

The last category, distributed or also called by agreement, entails all members contributing to key elaboration, with the key being a product of collective calculations, rather than a predetermined result. Typically, Diffie-Hellman-based, the security of such protocols is contingent on the adversary's computational capabilities.

Quantum cryptography emerges as a cornerstone application of quantum informatics, with various protocols developed for the needs of this specific field, including quantum key distribution (QKD) and quantum key agreement (QKA). When implemented correctly, QKD offers unconditional security. The BB84 protocol stands out as a prominent QKD example, aiming to resolve the classic key establishment problem by generating a symmetric key between two parties.

In contrast to classic methods, the security of quantum cryptography protocols, including QKD, is not contingent on the adversary's computational capabilities, but rather on quantum principles, enabling to detect intrusions during the key elaboration stage. Through QKD, multiple participants can securely share secret and random keys.

It is worth noting the distinction between QKD and QKA. In QKD, one entity generates the key distributed to other participants, while in QKA, none of the participants possess the key initially. Instead, it is collectively generated through communication and calculations, ensuring that no subset of participants can determine the key independently.

Since the pioneering work performed by Zhou *et al.* [6], numerous quantum key agreement protocols have been proposed, including two-party and multi-party variants (MQKA protocols). This study focuses specifically on multi-party protocols, categorized based on transmission structures into complete (CGT-MQKA), circle (CT-MQKA), and tree (TT-MQKA) configurations (Fig. 1).

In the complete-graph category, every participant shares their secret sequence with all other participants. Within the circle category, one participant transmits the sequence to the next participant, who then processes it and subsequently passes it on to the subsequent participant, continuing until the sequence returns to the original sender. The tree category involves a root participant disseminating information to other participants organized in a hierarchical tree structure.

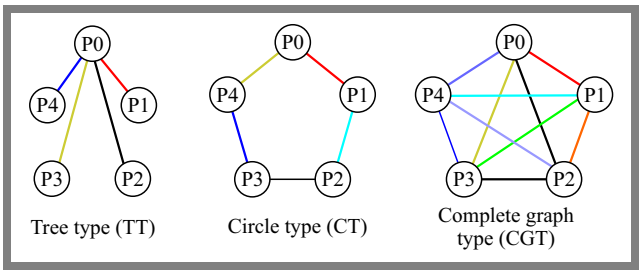

Fig. 1. Classification of quantum key agreement protocols [7].

3. Preliminaries

3.1. Superposition

In quantum computing, the fundamental unit of information infrastructure is represented as a qubit. The polarization state of a photon serves as the pivotal characteristic defining the value held by the qubit. From the physical perspective, the qubit may be represented by elementary entities, such as photons. In the context of quantum computing, the qubit plays the role of a data unit, analogous to the classical bit. Notably, a single qubit possesses the capacity to concurrently store two values of information (1 and 0). In a Dirac notation, the qubit is symbolized as follows:

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

where:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{and} \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \ . \tag{2}$$

In Eq. (1) α, β are complex numbers, where:

$$|\alpha|^2 + |\beta|^2 = 1 \ . \tag{3}$$

α represents the probability amplitude to have $|0\rangle$ and β is the probability amplitude to have $|1\rangle$ after the measurement.

3.2. Non-cloning Theorem

The qubit, in accordance with the principle of superposition, can concurrently embody two distinct values. Upon measurement, one of these values is arbitrarily discarded, giving way to the manifestation of the other. This inherent property renders replication of a quantum state unattainable, as any attempt to perform a measurement inevitably alters the state, thereby modifying the value held by the qubit. A fundamental tenet underpinning quantum cryptography is encapsulated in the non-cloning theorem.

3.3. Heisenberg Uncertainty Principle

A quantum state comprises a confluence of diverse parameters, such as the position and velocity of a photon. According to Heisenberg's principle, attempting to measure one of these parameters inherently disrupts the other. This principle fortifies the rationale behind the non-cloning theorem, as obtaining a complete description of a quantum state is imperative for replication. Conversely, any measurement inherently perturbs certain characteristics, precluding the precise duplication of the quantum state without possessing its comprehensive description.

These foundational principles afford security to communications shared between entities, often denoted as Alice and Bob, without susceptibility to eavesdropping. Any attempt by a third-party listener to intercept the communication inevitably distorts its value, creating a discrepancy discernible to Alice and Bob through a post-transmission examination of

error rates. This mechanism enables the detection of eavesdropping attempts, as alterations to the message's value are indicative of unauthorized intrusion. It is important to note that while quantum cryptography facilitates secure communication, the inherently random nature of the messages precludes the transmission of predetermined information. Nevertheless, the random string generated through quantum processes may serve as an ideal key in the implementation of the one-time pad (OTP) cryptographic protocol.

4. Related Works

In contrast to key distribution, the key agreement process involves the creation of a key through exchanges between two or more participants, in a manner analogous to the Diffie-Hellman protocol. Each participant contributes a part of the key, yet none can individually ascertain the key in its entirety. Such a concept was introduced by Zhou *et al.* in [6]. The paper presents the first proposal of a quantum key agreement protocol, leveraging quantum teleportation to generate shared keys. However, subsequent analysis revealed its susceptibility to insider attacks [8].

In [9], a key agreement protocol based on BB84 was introduced. Nevertheless, this protocol relied on quantum memory, the utilization of which remains prohibitively costly. Sun *et al.* [10] proposed enhancements to the multiparty quantum key agreement, introducing two additional unitary operations to improve protocol efficiency within the circle-based category. Another protocol predicated on entanglement swapping was proposed, and it utilized Bell states as quantum resources and Bell measurements as primary operations [11]. Furthermore, an alternative multi-party quantum key agreement protocol based on two entangled qubits was proposed, albeit it was applicable solely to three parties [12].

In article [13], leveraging a tree structure, the authors proposed a multiparty quantum key agreement protocol grounded in Greenberger-Horne-Zeilinger (GHZ) states. Subsequently, Ye *et al.* [14] and Sun *et al.* [15] developed two-party and three-party quantum key agreement protocols, respectively, based on unitary operations and four-qubit cluster states or an entangled six-qubit state.

In [16], the authors proposed a quantum key agreement protocol based on BB84. Differing from prior endeavors [8], this protocol ensures computational security against internal attacks through the utilization of hash functions, while preserving unconditional security against eavesdropping. It is intended for key agreement between two participants.

Table 1 summarizes the different protocols, referencing the best known solutions described in the literature.

To the best of our knowledge, there is no proposal for a key agreement protocol that may be used within a tree-organized group, founded on BB84 and employing XOR operation. This deficiency underscores our motivation to present protocols grounded in BB84, not only due to their well-established security but also their XOR-based efficiency, with simplicity and low execution cost being their additional advantages.

5. Proposed Protocol

The objective of the proposed tree-MQKA (T-MQKA) protocol is to address the issue of key generation and distribution in a hierarchically structured group relying on a tree configuration. The concept is based on the BB84 quantum key distribution protocol the security of which has been demonstrated in [17]. The initial phase of the protocol entails the execution of BB84 between the root node and each node within the group. Subsequent to the completion of the BB84 process, each node shares a confidential and secure string with the root node. Each of these strings is divided into two parts:

– seed (designated as S), which is employed in the generation of the final group key,

– part K, which serves as an intermediate key during the distribution of parts.

Upon completion of the initial stage, the root node proceeds with the party distribution phase. Subsequently, each node within the group will receive the S parts of the other nodes which have been XOR-ed. The resulting message is then encrypted with its K part and transmitted from the root. The aforementioned procedure is repeated for each remaining node in the group until all the nodes have obtained the S parts of the other nodes. Once a node has received the encrypted message containing the combination of S parts, it can decrypt the message and then add its own part, thereby obtaining the final group key.

5.1. Algorithm Description

For clarity, an algorithmic description of procedure is provided below.

Step 1 is the first phase, where the group is initialized and the root node is selected from among the nodes in the group, as shown in Algorithm 1.

Algorithm 1 Initialization of participants and the central node

1: Let N be the number of participants
2: Let $P[1], P[2], \ldots, P[N]$ be the set of participants
3: Let C be the central node selected from the participants

In step 2, each participant executes the BB84 protocol, thereby maintaining a chain that is shared with the root node (Algorithm 2). The security and confidentiality of the chain are guaranteed by the inherent security of BB84.

Algorithm 2 Execution of the BB84 protocol

1: **for** each participant $P[i]$ from the set of participants, where i ranges from 1 to N **do**
2: **if** $P[i] \neq C$ **then**
3: $P[i]$ executes the BB84 protocol with the central node C.
4: Store the resulting key as $K[i]$.
5: **end if**
6: **end for**

Tab. 1. Summary of existing quantum key agreement protocols.

Protocol name	Purpose	Key features	Reference
Quantum key agreement via teleportation	QKA	Utilizes quantum teleportation; susceptible to insider attacks	[6], [8]
BB84-based quantum key agreement	QKA	Relies on quantum memory; costly	[9], [16]
Enhanced multiparty quantum key agreement	MQKA	Introduces two additional unitary operations	[10]
Entanglement swapping-based multiparty QKA	MQKA	Uses Bell states and Bell measurements	[11]
Two qubits entanglement-based three-party QKA	3-party QKA	Applicable solely to three parties	[12]
GHZ states-based multiparty QKA	MQKA	Leverages a tree structure and GHZ states	[13]
Unitary operations-based two-party QKA	2-party QKA	Uses unitary operations and four-qubit cluster states	[14]
Unitary operations-based three-party QKA	3-party QKA	Utilizes an entangled six-qubit state	[15]
BB84-based two-party QKA with hashes	2-party QKA	Ensures computational security against internal attacks; hash functions	[8], [16]

Finally, in step 3, the chain is divided into two parts, the first of which is employed in the key agreement process, while the other is utilized as an intermediate key in the key distribution process. The final phase of the protocol entails the distribution of the key seeds and the construction of the final key through the assembly of all the key seeds. The chain obtained in the second phase is split into two parts: the first part is used in the key agreement process, while the other is used as an intermediate key in the key distribution process. The missing seeds S_i are transmitted to each node by sending an encrypted message containing them. Encryption is performed with the k part that it shares with the root – see Algorithm 3.

Algorithm 3 Key agreement and key distribution

1: $Xor\_result = 0$
2: **for** each participant $P[i]$ in the set of participants **do**
3: split $K[i]$ into $S_{C,P[i]}$ and $K_{C,P[i]}$
4: **end for**
5: **for** each participant $P[j]$ where $j \neq i$ **do**
6: $Xor\_result = Xor\_result \oplus S_{C,P[j]}$
7: $Xor\_result = Xor\_result \oplus K_{C,P[i]}$
▷ Central node encrypts the result
8: $C \rightarrow P[i] : Xor\_result$
▷ Central node sends encrypted result to $P[i]$
9: $Xor\_result = Xor\_result \oplus K_{C,P[i]}$
▷ $P[i]$ decrypts Xor_result
10: $Xor\_result = Xor\_result \oplus S_{C,P[i]}$
▷ $P[i]$ adds its seed and retrieves the final key
11: **end for**

Consider an example with a group of 4 nodes, a root node and 3 other nodes, as illustrated in Fig. 2, with the notation provided in Tab. 2. The center node A, acting as the initiator, shares the generated seeds of each participant with the remaining members through an XOR operation. Subsequently, the root node encrypts the XOR output using a shared key with each participant. Upon receiving the encrypted seeds, each participant (B, C, and D) decrypts the data and incorporates its own seed to derive the final key. This iterative process is carried out for all the members. At the end, every participant will have obtained the final key.

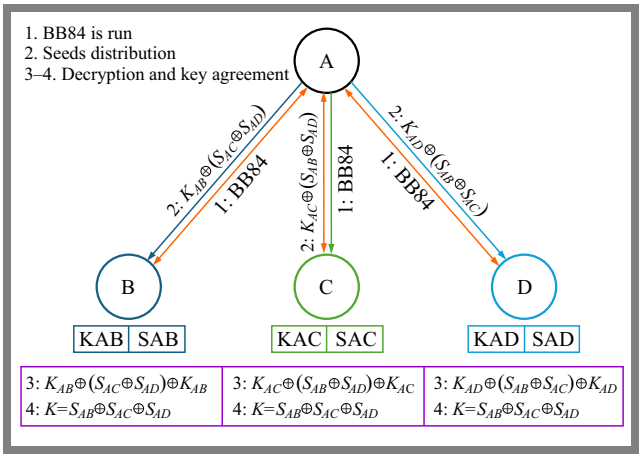

Fig. 2. New T-MQKA based on BB84.

Tab. 2. Notations of the example of proposed protocol.

Notation	Description
A, B, C, D	Participants (nodes)
K_{AB}, K_{AC}, K_{AD}	Encoding keys of participants B, C, and D
S_{AB}, S_{AC}, S_{AD}	Seeds part shared between A and respectively B, C, and D
$A \rightarrow B$:	A sends to B
$\oplus$	Bitwise XOR (exclusive OR)

The step-by-step equations shown in Algorithm 4 illustrate the communication process between A, B, C, and D, respectively, aiming to determine final key.

Algorithm 4 Key agreement and key distribution

1: A $\rightarrow$ B : $(S_{AC} \oplus S_{AD}) \oplus K_{AB}$
▷ A sends $(S_{AC} \oplus S_{AD})$ encrypted by K_{AB}
2: B does $[(S_{AC} \oplus S_{AD}) \oplus K_{AB}] \oplus K_{AB} \oplus S_{AB}$
▷ B decrypts and adds its seed S_{AB} and gets the final key
3: A $\rightarrow$ C : $(S_{AB} \oplus S_{AD}) \oplus K_{AC}$
▷ A sends $(S_{AB} \oplus S_{AD})$ encrypted by K_{AC}
4: C does $[(S_{AB} \oplus S_{AD}) \oplus K_{AC}] \oplus K_{AC} \oplus S_{AC}$
▷ C decrypts and adds its seed S_{AC} and gets the final key
5: A $\rightarrow$ D : $(S_{AB} \oplus S_{AC}) \oplus K_{A,D}$
▷ A sends $(S_{AB} \oplus S_{AC})$ encrypted by K_{AD}
6: D does $[(S_{AB} \oplus S_{AC}) \oplus K_{AD}] \oplus K_{AD} \oplus S_{AD}$
▷ D decrypts and adds its seed S_{AD} and gets the final key

5.2. Protocol Properties

Tree topology is selected based on the fact that it is more resilient to collusive attacks, as it needs fewer qubits for the all-to-all transmission of quantum states. Consequently, it reduces the need for quantum resources. It is worth noting another advantage here: if a device operating within a tree topology fails, the failure does not necessarily affect the entire communication network, making it more reliable.

The proposed solution exhibits the following characteristics:

- Implementation of the XOR operation in the protocols lowers computational complexity.
- The BB84 protocol is used, relying on a single-photon scenario, making the proposed solution compliant with current technology.
- Security is based on the proven capabilities of BB84 [17], [18] which employed probabilistic model checking techniques with the PRISM model checker. The resultant key derived from BB84 is safeguarded by the principles of quantum physics, automatically detecting any eavesdropping attempts during its generation phase. Similarly, the seed, being both secure and random, remains undisclosed until execution of the BB84 protocol.
- There is no need to preload any of the secret or random seeds, as they are secured by quantum rules.
- Generation of the group's common key takes place through secure exchanges between all participants, with each of them contributing their seed to formulate the final key, and such an approach allows to keep it unknown to any participants beforehand.
- Participants are prevented from recovering the shares of other participants, is realized by XOR operations.
- Thanks to optimization of the message exchange process, the solution requires only $2\ (N-1)$ messages.
- The participating nodes contribute equally to the generation of the final key. Each node adds its seed to the collective construction of the group key.

6. Security Analysis

The concept of agreeing upon a quantum key presents distinct challenges compared to key distribution, as it necessitates active involvement of all participants in the key generation process. Unlike key distribution, where a single participant prepares and disseminates the key to others, quantum key agreement demands that no participant possesses or is capable of predicting the key beforehand. It requires each participant to contribute equally to the key generation process, ensuring that the resulting key remains unpredictable to any subset of participants.

The proposed tree quantum key agreement protocol (T-MQKA) upholds this fundamental principle by ensuring that participants lack prior knowledge of the key. Instead, each participant contributes a seed to the key generation process, relying on a secure exchange facilitated by keys generated via the BB84 protocol. This quantum-based methodology guarantees the complete security of the exchanged keys, thereby preserving the integrity and fairness of the key generation process in the quantum key agreement phase.

6.1. Immunity to Internal Attacks

Internal attacks can be initiated by malicious participants within the network. These attacks are divided into two main categories: attacks undertaken by a single participant and massive attacks involving multiple participants.

A single malicious participant may attempt to deduce the group key using their own key $K_{C,P[i]}$ and the messages received from the root node. The resilience of T-MQKA to such attacks is ensured through the following mechanisms:

- Key division. Each BB84 key generated by the root node is divided into two parts: S and $K_{C,P[i]}$. The S part is used to generate the group key, while the $K_{C,P[i]}$ part is used for encrypting and decrypting S. This division adds a layer of complexity, as a single participant would need to know multiple S and $K_{C,P[i]}$ values to deduce the group key.
- XOR operations. The m' messages sent from the root node to each participant are formed by XOR-ing multiple S values with the participant's $K_{C,P[i]}$. For example, $m' = S_{i+1} \oplus S_{i+2} \oplus S_n \oplus (K_{C,P[i]})$. The receiving participant can then compute the group key by XOR-ing m' with its $K_{C,P[i]}$. This XOR operation ensures that knowing a single $K_{C,P[i]}$ value without the corresponding S values provides no useful information.

Therefore, it is mathematically challenging for a single participant to reconstruct the group key without possessing the knowledge of other S and K values, which makes the protocol resistant to single-participant attacks.

In a collusive attack, multiple participants might collude to share their information to determine the group key. Despite this, T-MQKA remains secure due to the following reasons:

- Multiple XOR-ed values. Each m' message sent to a participant includes XOR operations with multiple S values from different nodes. For example, $m' = S_i \oplus S_{i+2} \oplus S_n \oplus (K_{C,P[i]}$ for $N_{i+1})$. This means that even if two or

more participants collude, they need to gather all relevant S values from other nodes to reconstruct the group key.

- Information distribution. The information necessary to compute the group key is distributed across all participants. Each participant has only partial information, and the full reconstruction requires knowledge of all S and $K_{C,P[i]}$ values, which is highly unlikely to be achieved through collusion alone.

Thus, the complexity and distribution of the S and $K_{C,P[i]}$ values in the protocol ensure its robustness and resilience to such attacks.

6.2. External Attacks

External attacks are carried out by entities that are not part of the network, such as eavesdroppers or hackers attempting to intercept communications. T-MQKA incorporates several mechanisms to safeguard against these threats.

An external attacker might try to intercept the m' messages sent from the root node to other nodes. The protocol protects against this type of attack through the following measures:

- Encrypted messages. Each m' message is encrypted using a unique $K_{C,P[i]}$ key. Intercepting these messages without the corresponding $K_{C,P[i]}$ key provides no useful information. The attacker would only see the result of multiple XOR operations, appearing as random data without the proper keys.
- Integrity checks. Any attempt to modify the intercepted messages would be detected during decryption by the recipient nodes. The XOR operation ensures that any tampering results in the computation of an incorrect group key, which would be immediately noticeable to the participants.

An attacker might attempt to replace the $K_{C,P[i]}$ or S keys during transmission. The protocol is immune to this threat through:

- Quantum security properties. The S keys are distributed using the BB84 protocol, ensuring that any interception or replacement attempt is immediately detectable due to the principles of quantum mechanics. Any eavesdropping will introduce detectable errors in the key.
- Detection mechanisms. Participants can detect any anomalies in the keys through standard BB84 error rate checks. If the error rate exceeds a predefined threshold, the participants know that an interception attempt has occurred.

Half of the transmitted particles are chosen as detection particles to prevent eavesdropping. These detection particles serve as a safeguard using two mechanisms described below:

- Random selection. The selection of detection particles is random, making it difficult for an external attacker to predict which particles are used for detection. This randomness ensures that any eavesdropping attempts are likely to be detected.
- Error rate monitoring. By monitoring the error rates of the detection particles, the protocol may identify and thwart any eavesdropping attempts. High error rates indicate the presence of an eavesdropper, warning the participants.

In conclusion, T-MQKA offers robust security against both internal and external attacks. The use of quantum properties for key distribution, combined with complex operations involving S and $K_{C,P[i]}$ keys, ensures that the group key remains secure and that any attempts to compromise the key are quickly detected. The protocol's design inherently protects it against various threat vectors, maintaining the integrity and confidentiality of the shared keys.

7. Qubit Efficiency Calculation

Qubit efficiency (QE) is a key measure of how effectively quantum resources are used in the protocol. It is defined as the ratio of the total bits in the final group key to the total number of qubits used while executing the protocol, including any classic bits exchanged for decoding the message, as described in [19]. In the proposed protocol, qubit efficiency is determined by the following factors:

- Each BB84 key is divided into two parts: $S_{C,P[i]}$, which helps form the group key and $K_{C,P[i]}$, used for encryption and decryption. For the purpose of our this analysis, $S_{C,P[i]}$ is assumed to be half the length of the BB84 key (n), hence $|S_{C,P[i]}| = \frac{n}{2}$.
- The total length of the group key is equal to $\frac{n}{2}$.
- The generation of a BB84 key between the central node and each child node requires n qubits, and the total number of qubits used in the network is $(N-1) \times n$.

Based on these assumptions, the qubit efficiency for a network with N participants is given by:

$$QE = \frac{c}{q+b}\,, \tag{4}$$

where c represents the length of the final group key, which is $\frac{n}{2}$, q is the total number of qubits used, calculated as $(N-1) \times n$, and b is the total number of classical bits exchanged, which is considered negligible in this analysis.

The qubit efficiency equation simplifies to:

$$QE = \frac{\frac{n}{2}}{(N-1) \times n} = \frac{1}{2\,(N-1)}\,. \tag{5}$$

Equation (5) illustrates that qubit efficiency decreases as the number of nodes in the network increases. It highlights the protocol's ability to use quantum resources effectively, while maintaining security, particularly in large networks.

Tab. 3. Comparative analysis of MQKA protocols in a tree topology.

Ref.	Topology	Quantum efficiency	Security
[20]	Tree	$QE = \frac{1}{11N}$	Secure
[21]	Tree	$QE = \frac{1}{N(2^{N-1})}$	Secure
Proposed	Tree	$\frac{1}{2\,(N-1)}$	Secure

Overall, this analysis demonstrates the effectiveness of the proposed T-MQKA protocol in optimizing quantum resource utilization, while simultaneously ensuring robust security. The balance between resource efficiency and security makes our protocol well-suited for secure communications relying on quantum networks.

8. Comparative Analysis of Different MQKA Protocols

The comparison of several QKA protocols presented in Tab. 3 shows the quantum efficiency of the proposed T-MQKA protocol and that of different existing MQKA protocols, while Fig. 3 contains a graph illustrating the quantum efficiency of different QKA protocols, including the proposed solution, TT-MQKA, and [21]. The x–axis of the graph represents the number of participants N, while the y–axis indicates the quantum efficiency level.

Figure 3 clearly shows that the new protocol offers decent quantum efficiency compared to other solutions, particularly as the number of nodes (participants) increases, highlighting its potential for more robust and scalable quantum key agreement applications.

The heatmap presented in Fig. 4 visualizes the fact that the proposed T-MQKA protocol performs better, in terms of quantum efficiency, than TT-MQKA and [21]. Such a result shows that the new T-MQKA approach is resource-efficient, meaning that it should be taken into account by all parties as the most competent quantum key distribution solution suitable for small and large-scale networks. The results of fault tolerance tests show that the proposed protocol consistently achieves high QE ratings, regardless of the count of nodes.

9. Conclusion

In this paper, we introduced a tree multiparty quantum key agreement protocol (T-MQKA) intended for securing communications. It is based on the principles of the BB84 protocol and facilitates secure key agreement among multiple participants organized in a tree topology. This new protocol has demonstrated decent quantum efficiency compared to other protocols in its category. Moreover, security analysis has confirmed T-MQKA's resilience against a multitude of potential attacks, ensuring the integrity and confidentiality of data transmission. Its robustness covers both external and internal threats, underscoring the protocol's reliability in safeguarding sensitive information.

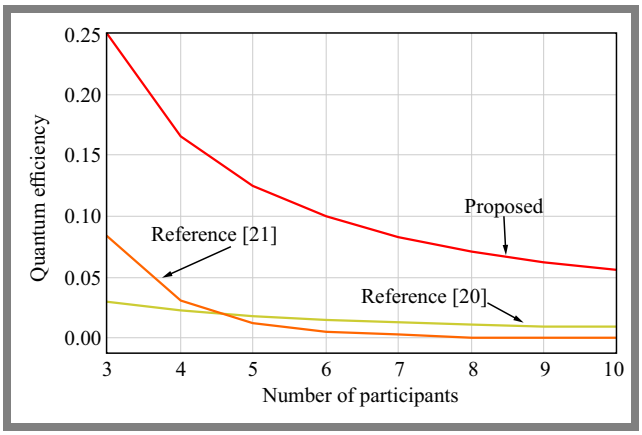

Fig. 3. Quantum efficiency comparison of different T-MKQA protocols.

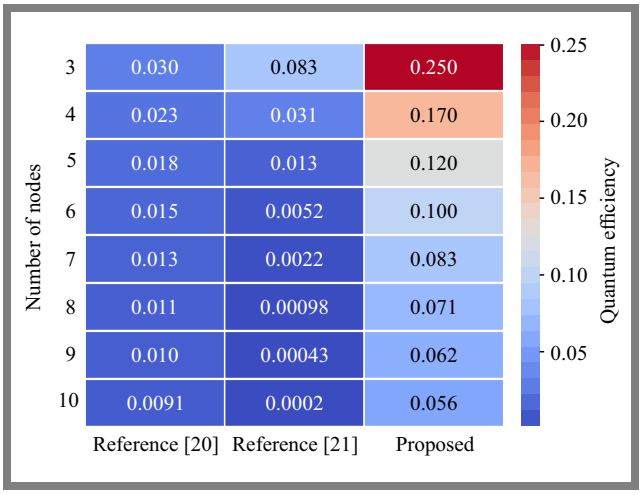

Fig. 4. Heatmap visualization of QE of different tree MKQA protocols for $N = 3 \ldots 10$.

References

[1] Y. Shen, Z. Sun, and T. Zhou, "Survey on Asymmetric Cryptography Algorithms", *2021 International Conference on Electronic Information Engineering and Computer Science (EIECS)*, Changchun, China, 2021 (https://doi.org/10.1109/EIECS53707.2021.9588106).

[2] Y. Cheng, Y. Liu, Z. Zhang, and Y. Li, "An Asymmetric Encryption-based Key Distribution Method for Wireless Sensor Networks", *Sensors*, vol. 23, no. 14, 2023 (https://doi.org/10.3390/s23146460).

[3] V. Martin, J. Martinez-Mateo, and M. Peev, "Introduction to Quantum Key Distribution", in: *Wiley Encyclopedia of Electrical and Electronics Engineering*, 2017 (https://doi.org/10.1002/047134608X.W8354).

[4] B.-X. Liu, R.-C. Huang, Y.-G. Fang, and G.-B. Xu. "Measurement-device-independent Multi-party Quantum Key Agreement", *Frontiers in Quantum Science and Technology*, vol. 2, 2023 (https://doi.org/10.3389/frqst.2023.1182637).

[5] Y. Challal and H. Seba, "Group Key Management Protocols: A Novel Taxonomy", *International Journal of Computer and Information Engineering*, vol. 2, no. 1, 2008 (https://doi.org/10.5281/zenodo.1077968).

[6] N. Zhou, G. Zeng, and J. Xiong, "Quantum Key Agreement Protocol", *Electronics Letters*, vol. 40, no. 18, pp. 1149–1150, 2004 (https://doi.org/10.1049/el:20045183).

[7] B. Liu, D. Xiao, H.-Y. Jia and R.-Z. Liu, "Collusive Attacks to 'Circle-type' Multi-party Quantum Key Agreement Protocols", *Quantum Information Processing*, vol. 15, pp. 2113–2124, 2016 (https://doi.org/10.1007/s11128-016-1264-5).

[8] K.-F. Yu *et al.*, "Design of Quantum Key Agreement Protocols with Fairness Property", arXiv, 2015 (https://doi.org/10.48550/arXiv.1510.02353)

[9] S.-K. Chong and T. Hwang, "Quantum Key Agreement Protocol Based on BB84", *Optics Communications*, vol. 283, no. 6, pp. 1192–1195, 2010 (https://doi.org/10.1016/j.optcom.2009.11.007).

[10] Z. Sun *et al.*, "Improvements on Multiparty Quantum Key Agreement with Single Particles", *Quantum Information Processing*, vol. 12, pp. 3411–3420, 2013 (https://doi.org/10.1007/s11128-013-0608-7).

[11] R.-H. Shi and H. Zhong, "Multi-party Quantum Key Agreement with Bell States and Bell Measurements", *Quantum Information Processing*, vol. 12, pp. 921–932, 2013 (https://doi.org/10.1007/s11128-012-0443-2).

[12] X.-R. Yin, W.-P. Ma, and W.-Y. Liu, "Three-party Quantum Key Agreement with Two-photon Entanglement", *International Journal of Theoretical Physics*, vol. 52, no. 11, pp. 3915–3921, 2013 (https://doi.org/10.1007/s10773-013-1702-4).

[13] G.-B. Xu, Q.-Y. Wen, F. Gao, and S.-J. Qin, "Novel Multiparty Quantum Key Agreement Protocol with GHZ States", *Quantum Information Processing*, vol. 13, p. 2587–2594, 2014 (https://doi.org/10.1007/s11128-014-0816-9).

[14] Y.-F. He and W. Ma, "Quantum Key Agreement Protocols with Four-qubit Cluster States", *Quantum Information Processing*, vol. 14, no. 9, pp. 3483–3498, 2015 (https://doi.org/10.1007/s11128-015-1060-7).

[15] Z. Sun *et al.*, "Multi-party Quantum Key Agreement by an Entangled Six-qubit State", *International Journal of Theoretical Physics*, vol. 55, pp. 1920–1929, 2016 (https://doi.org/10.1007/s10773-015-2831-8).

[16] P. Wang, R. Zhang, and Z. W. Sun, "Practical Quantum Key Agreement Protocol Based on BB84", *Quantum Information and Computation*, vol. 22, pp. 241–250, 2022 (https://doi.org/10.26421/QIC22.3-4-3).

[17] P.W. Shor and J. Preskill, "Simple Proof of Security of the BB84 Quantum Key Distribution Protocol", *Physical Review Letters*, vol. 85, no. 2, p. 441–444, 2000 (https://doi.org/10.1103/PhysRevLett.85.441).

[18] M. Elboukhari, M. Azizi, and A. Azizi, "Verification of Quantum Cryptography Protocols by Model Checking", *International Journal of Network Security and Its Applications*, vol. 2, no. 4, pp. 43–53, 2010 (https://doi.org/10.5121/ijnsa.2010.2404).

[19] A. Cabello, "Quantum Key Distribution in the Holevo Limit", *Physical Review Letters*, vol. 85, pp. 5635–5638, 2001 (https://doi.org/10.1103/PhysRevLett.85.5635).

[20] H. Yang *et al.*, "A Tree-type Multiparty Quantum Key Agreement Protocol Against Collusive Attacks", *International Journal of Theoretical Physics*, vol. 62, art. no. 7, 2023 (https://doi.org/10.1007/s10773-022-05265-w).

[21] J. Gu and T. Hwang, "Improvement of Novel Multiparty Quantum Key Agreement Protocol with GHZ States", *International Journal of Theoretical Physics*, vol. 56, pp. 3108–3116, 2017 (https://doi.org/10.1007/s10773-017-3478-4).

Rima Djellab, Ph.D.
Computer Science Department
https://orcid.org/0000-0001-8072-1868
E-mail: r.djellab@univ-batna2.dz
LAMIE Laboratory, University of Batna 2, Batna, Algeria
http://www.univ-batna2.dz

Youssouf Achouri, Ph.D. Student
Computer Science Department
https://orcid.org/0009-0003-9500-496X
E-mail: y.achouri@univ-batna2.dz
LASTIC Laboratory, University of Batna 2, Batna, Algeria
http://www.univ-batna2.dz

Malak Emziane, Student
Renewable Energies and New Technologies Department
https://orcid.org/0009-0001-1485-8169
E-mail: emziane.malak@hns-re2sd.dz
LEREESI Laboratory, HNS-RE2SD, Batna, Algeria
http://www.hns-re2sd.dz

Lyamine Guezouli, Ph.D.
Renewable Energies and New Technologies Department
https://orcid.org/0000-0002-7406-7633
E-mail: lyamine.guezouli@hns-re2sd.dz
LEREESI Laboratory, HNS-RE2SD, Batna, Algeria
http://www.hns-re2sd.dz