Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/TQE.2020.DOI

Security and Fairness in Multi-Party Quantum Secret Sharing Protocol

ALESSIO DI SANTO[1], (Graduate Student Member, IEEE), WALTER TIBERTI [1], (Member, IEEE), AND DAJANA CASSIOLI[1], (Senior Member, IEEE)

[1]Department of Information Engineering, Computer Science, and Mathematics (DISIM), University of L'Aquila, 67100 L'Aquila, Italy

Corresponding author: Alessio Di Santo (email: alessio.disanto@graduate.univaq.it).

ABSTRACT *Quantum secret sharing* (QSS) is a cryptographic protocol that leverages quantum mechanics to distribute a secret among multiple parties. With respect to the classical counterpart, in QSS the secret is encoded into quantum states and shared by a *dealer* such that only an authorized subsets of participants, i.e., the *players*, can reconstruct it. Several state-of-the-art studies aim to transpose classical Secret Sharing into the quantum realm, while maintaining their reliance on traditional network topologies (e.g., star, ring, fully-connected) and require that all the n *players* calculate the secret. These studies exploit the Greenberger-Horne-Zeilinger (*GHZ*) state, which is a type of maximally entangled quantum state involving three or more qubits. However, none of these works account for redundancy, enhanced security/privacy features or authentication mechanisms able to fingerprint players. To address these gaps, in this paper we introduce a new concept of QSS which leans on a generic distributed quantum-network, based on a threshold scheme, where all the *players* collaborate also to the routing of quantum information among them. The *dealer*, by exploiting a custom flexible weighting system, takes advantage of a newly defined quantum Dijkstra algorithm to select the most suitable subset of t *players*, out of the entire set on n players, to involve in the computation. To fingerprint and authenticate users, *CRYSTAL-Kyber* primitives are adopted, while also protecting each *player*'s privacy by hiding their identities. We show the effectiveness and performance of the proposed protocol by testing it against the main classical and quantum attacks, thereby improving the state-of-the-art security measures.

INDEX TERMS Quantum algorithm, Quantum circuits, Quantum communications, Quantum computing, Quantum cryptography, Quantum entanglement, Quantum key distribution, Quantum networks.

I. INTRODUCTION

SECRET SHARING encompasses methodologies for distributing a secret among a group of individuals, each of whom does not possess any comprehensible information about it. Only when a requisite number of participants combine their respective shares, the original secret can be reconstructed. In contrast to insecure secret sharing, where an attacker can incrementally acquire more information with each share, *secure secret sharing* adheres to an 'all or nothing' principle, where 'all' denotes the necessary number of shares needed to reconstruct the secret. Secret reconstruction can follow two distinct schemes. The first is the (n, n) scheme, which relies on full participation, meaning every participant holds a piece of the secret, and it can only be recovered when all pieces are combined. The second is the (t, n) scheme, which introduces a threshold, t, allowing a subset of participants (of size t) to reconstruct the secret without the involvement of the entire group. Major studies on secret sharing began in 1979 by G. R. Blakley and Adi Shamir. Blakley's research employs hyperplane geometry to address the secret sharing problem. To create a (t, n) threshold scheme, each of the *n* participants (2 or more) are provided with a hyperplane equation within a t-dimensional space over a finite field [1]. Shamir's scheme, on the other hand, relies on a similar concept but is based on polynomial interpolation [2].

Since 1997, when Peter Shor demonstrated how modern mathematical pillars about hard-to-solve problems could be easily defeated by moving into the quantum realm [3], researchers have felt the need to find additional security measures in the same domain that defeated them, i.e., the quantum one. Hence, the number of studies on *Quantum Secret Sharing* [4], [5] and *Quantum Fairness* [6], [7] grew since then, with numerous protocols addressing dynamic

participation and secure quantum information distribution.

The first class of QSS-protocols were explored in 1999, by exploiting *Greenberger–Horne–Zeilinger* states [8], which is a type of multi-particle entangled quantum state that plays a central role in quantum information theory, especially in studies of quantum entanglement and nonlocality [9].

Following this seminal paper, Jia *et al.* [10] introduced *dynamic QSS* using *GHZ states*, while Liao *et al.* [11] focused on ring-based topologies. Other notable works include hierarchical *QSS* protocols [12], [13], d-dimensional GHZ-based sharing [14], and single-photon approaches [15]. Nowadays, these kinds of studies focus on improving the security and efficiency of their solutions, while addressing their vulnerabilities. However, there are still open challenges to address, including the fault tolerance, distributed and flexible routing schemes, authentication methods for player privacy and security.

In this paper, we propose a new QSS protocol suite to address these challenges in a *dealer-players* scenario based on a generic *Entanglement-based Quantum Secret Sharing Protocol* over an adaptive quantum network architecture. Our approach leverages a newly defined *Quantum-Dijkstra* algorithm (correlated with a custom *weighting system*), *QKD*, *CRYSTALS-Kyber*, and the properties of *fairness* and the *CIA Triad (Extended)*.

1) Paper's Contribution

The core contributions of our research can be summarized as follows:

- Introduction of dynamic and flexible network topologies modeled on distributed computation.
- Enhanced player privacy by restricting network topology knowledge to reduce collusion.
- Development of a Quantum-Dijkstra algorithm for optimal participant selection.
- Establishment of a custom weighting system, which relies on both classical and quantum parameters, to support Quantum-Dijkstra's usage.
- Integration of CRYSTALS-Kyber for continuous post-quantum player authentication.
- Extension of fairness into a *(t,n)*-scheme for tamper detection.
- Implementation of the extended CIA Triad Framework for a comprehensive information security.

The rest of the paper is organized as follows. In Section II, a comparison with the state-of-the-art is proposed; in Section III the proposed system model is shown. Section IV describes the proposed protocol, while Section V presents the validation of the proposed protocol, by a numerical example in Section V-A, and by Qiskit simulations in Section V-B to compare the protocol's performance in two case studies, i.e., in a locally simulated environment and over the IBM Quantum Network. Additional considerations on our protocol's *fairness* and *CIA Triad (extended* properties are discussed and analyzed in Section V-C and Section V-D. Final conclusions are presented in Section VI.

II. RELATED WORK

Quantum Secret Sharing (QSS) protocols, rooted in the foundational work by Hillery *et al.* [8], aim to securely distribute a secret among multiple parties such that only authorized subsets can reconstruct it. These protocols have evolved along two primary paradigms: those relying on entangled states and those employing *Mutually Unbiased Bases* (MUB). Each approach has contributed distinct advantages while also opening pathways for further studies to enhance scalability, fault tolerance, privacy, and authentication.

Several studies have sought to enhance QSS through the use of entangled quantum states. For example, the work by Jia *et al.* [10] introduced dynamic QSS leveraging *Greenberger-Horne-Zeilinger* (GHZ) states, enabling flexible participation. Similarly, Liao *et al.* [11] explored ring-based network topologies to simplify implementation, while Qin *et al.* [14] extended the model to d-dimensional GHZ states to enhance scalability. Building on these concepts, Hsu *et al.* [12] and Mishra *et al.* [13] introduced hierarchical dynamic QSS protocols, facilitating group-based secret sharing with improved flexibility. While these contributions advanced the field significantly, exploring redundancy mechanisms and privacy-preserving techniques in such dynamic environments remains a promising direction for future research.

Another notable body of work centers on MUB-based QSS schemes. Lu *et al.* [4], for instance, proposed an entanglement-free protocol that integrates verification states to detect eavesdropping. This framework ensures information-theoretic security and provides a solid foundation for secure QSS. Nevertheless, resource efficiency and scalability in noisy or dynamic quantum networks present promising directions for future investigations. Priyanka *et al.* [16] combined MUB principles with the Quantum Fourier Transform for participant verification, offering a novel perspective that invites additional exploration into robust post-quantum authentication mechanisms to address emerging threats.

Zhou *et al.* [17] and Gong *et al.* [18] have made significant contributions to semi-quantum private comparison (SQPC) protocols. Zhou *et al.* introduced a multi-party protocol using d-level GHZ states, enhancing scalability and efficiency through high-dimensional entanglement. Gong *et al.*, on the other hand, proposed a Bell state-based protocol that simplifies implementation for near-term quantum devices. While these protocols demonstrate strong foundations, some challenges remain in extending their applicability to larger-scale and dynamically changing networks. Zhou *et al.*'s protocol, while scalable, allows for further studies on dynamic participant selection and adaptability to real-world network conditions. Gong *et al.*'s approach, though simpler to implement, was not designed for scenarios requiring high scalability or resilience to noise. These aspects present opportunities for further exploration and enhancement.

Dynamic QSS schemes have also gained significant attention in recent years. Hu *et al.* [19] proposed a novel

approach in high-dimensional quantum systems to improve scalability, while You *et al.* [15] explored single-photon-based QSS schemes between multiple parties. Song *et al.* [20] investigated generalized GHZ states in star topology networks, highlighting the potential for simplified implementation. Similarly, Yang *et al.* [21] and Tian *et al.* [22] improved multi-party QSS protocols by incorporating enhanced routing strategies and centralized architectures, respectively. These studies showcase various ways to optimize QSS for dynamic environments while also encouraging further exploration into decentralized architectures and adaptive routing mechanisms.

Addressing fairness in QSS has also been a focal point of research. Liu *et al.* [7] pioneered a fairness-enhanced (n, n)-threshold protocol that ensures equitable outcomes for all players, introducing mechanisms to detect and mitigate cheating. Similarly, Kang *et al.* [23] applied fairness principles to continuous-variable QSS, opening avenues for examining how adaptive routing and player authentication could further improve performance in changing network conditions. Tian *et al.* [22] introduced fairness mechanisms in their Bell-state-based multi-party QSS scheme. Additional opportunities to extend fairness into more complex network configurations, not fixed to standard topologies and developed over real-world operative conditions arise.

More recent studies have attempted to bridge these aforementioned gaps by integrating advanced cryptographic techniques. For example, Li *et al.* [24] introduced a trusted third-party system to verify player identities and to manage entanglement distribution, while Schauer *et al.* [25] explored security checks for n-qubit systems to detect eavesdropping. However, these innovative contributions still set the stage for further examination of how decentralized architectures and fault tolerance mechanisms might enhance scalability and practical deployment in large, distributed quantum networks.

Overall, current QSS protocols provide a robust foundation for secure quantum communication but leave exciting opportunities for future research. Investigating redundancy to enhance fault tolerance, privacy-preserving measures, and adaptive routing strategies could further enrich this domain. Moreover, exploring scalable distributed architectures and integrating post-quantum authentication frameworks would enhance their applicability in high-security scenarios.

To address these opportunities, our work introduces a novel QSS protocol designed for distributed quantum networks. By leveraging a Quantum-Dijkstra algorithm and a flexible weighting system, it enables dynamic participant selection and adaptive routing, ensuring fault tolerance and scalability. The integration of *CRYSTAL-Kyber* primitives provides post-quantum authentication, enhancing player privacy and protecting against quantum adversaries. Furthermore, our protocol extends fairness principles to (t, n)-threshold schemes and incorporates mechanisms to ensure secure retransmissions, aligning with the requirements of real-world quantum networks. These advancements collectively position our approach as a significant step forward in the development of practical and secure QSS systems.

A comparison of the main performance metrics of the proposed protocol vs. existing *QSS* schemes is presented in Table (1)). This comparison highlights that our protocol better addresses the following performances metrics:

- *Redundancy*, that in QSS ensures fault tolerance, enabling the protocol to remain functional even in the presence of failures, such as the loss of quantum channels or participant nodes.
- *Scalability*, which refers to the protocol's ability to efficiently accommodate an increasing number of participants or nodes while maintaining its performance.
- *Robustness*, that measures the protocol's resistance to noise, eavesdropping, and adversarial conditions, both classical and quantum.

III. SYSTEM MODEL

We consider a Quantum-Network which involves $(n-1)$ *players*, and a single *dealer*, where all of them are equipped with quantum devices connected via optical fibers in the network topology shown in Fig.1.

The *dealer* acts as a *Certification Authority* for the entire network and all participants are requested to register themselves to be authenticated. In the context of quantum networks, employing a distributed network topology rather than traditional structures like ring, common bus, or star configurations offers substantial advantages. Distributed topologies are characterized by nodes that relay information through a series of intermediate nodes rather than directly communicating with a central server.

In a distributed network, each node (or player) only needs to establish a connection with a single other node to become part of the network. This approach significantly enhances flexibility and scalability. When a new player joins the network, they simply connect to one existing player, and this new connection integrates this node into the network's communication fabric. This method avoids the bottleneck and single point of failure issues associated with centralized topologies, where every node must communicate directly with a central server.

In quantum networks, where the secure distribution of quantum information is paramount, relying on hops to reach a server through a distributed network can be more practical. This topology reduces the amount of direct communications needed between each node and the central server, thereby minimizing the potential for congestion and improving overall efficiency. Additionally, a distributed network allows for dynamic reconfiguration as nodes join or leave, making it more adaptable to real-world conditions compared to static topologies. This adaptability is crucial in quantum networks, where the integration of new participants and the maintenance of secure, reliable communication channels are essential.

TABLE 1: Comparison of QSS Protocols in Terms of Redundancy, Topology, Scalability, Routing Efficiency, Authentication, and Robustness.

Protocol	Topology	Redundancy	Scalability	Routing Efficiency	Authentication	Robustness
Jia et al. [10]	Star	No	Moderate	Basic (GHZ-based)	None	Limited
Liao et al. [11]	Ring	No	Limited	Fixed (Ring Topology)	None	Moderate
Hsu et al. [12]	Centralized	No	Moderate	Limited	Classical Methods	Moderate
Mishra et al. [13]	Hierarchical	No	Moderate	Group-Based (Hierarchical)	Classical Methods	Moderate
Qin et al. [14]	Centralized (GHZ)	No	High	d-dimensional GHZ states	None	Limited
You et al. [15]	Centralized	No	High	Basic (Single-photon)	None	Limited
Hu et al. [19]	Centralized (High-D)	No	High	Centralized (High-Dimensional)	None	Moderate
Gao et al. [26]	Hybrid	No	High	Advanced (Multi-party GHZ)	None	Moderate
Song et al. [20]	Star	No	Moderate	Fixed	None	Limited
Yang et al. [21]	Centralized	No	High	Improved Multi-party	Classical Methods	Moderate
Tian et al. [22]	Centralized (GHZ)	No	High	Centralized (GHZ-based)	None	Moderate
Proposed	**Distributed**	**Yes**	**High**	**Distributed (Quantum Dijkstra)**	**CRYSTAL-Kyber**	**Strong**

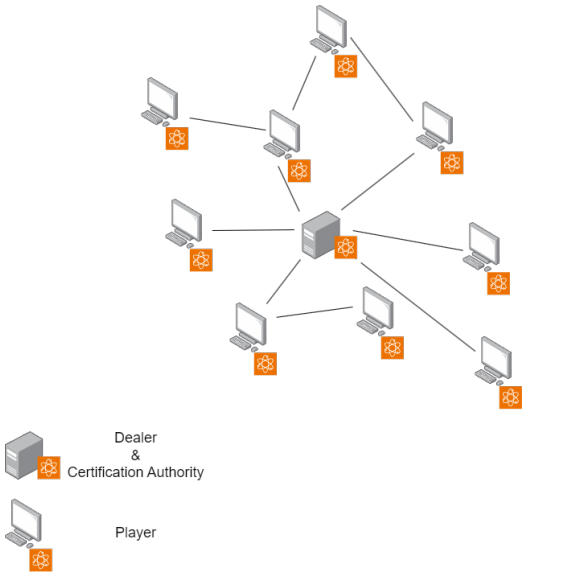

FIGURE 1: A Quantum-Network Topology for Secret Sharing.

IV. PROPOSED QUANTUM SECRET SHARING PROTOCOL

Inspired by [4] and [16], the proposed protocol allows the *dealer* to provide partial-secrets to a subset of t players out of n, where one of them will be able to correctly reconstruct the secret by exploiting the following operators:

- *Symmetric Polynomials*: These are polynomials in several variables that remains unchanged under any permutation of their variables [27];
- *Generalized Pauli Operators*: These are an extension of the traditional Pauli matrices, designed to operate on higher-dimensional quantum systems, known as qudits and fundamental in describing operations in d-dimensional Hilbert spaces. In a quantum system with d dimensions. A key characteristic of these operators is their mutual commutation relation, where applying the operators in different sequences introduces a phase factor based on the dimensionality of the system. This commutation property plays a critical role in defining the algebra of these operators [28].

To guarantee a (t, n)-threshold, the *dealer* first generates a symmetric polynomial given by:

$$\begin{aligned}\mathbf{G}(x,y) &= \sum_{i=0}^{t-1}\sum_{j=0}^{t-1} a_{ij}x^i y^j \\ &= a_{00} + a_{10}x + a_{01}y + \cdots + a_{t-1,t-1}x^{t-1}y^{t-1}\end{aligned} \tag{1}$$

with $\mathbf{G}(x,y) \in \mathbf{Z}_d$, where d is the prime field size used for modular arithmetic, $d \in \mathcal{Z} \mid d \bmod 2 \neq 0 \wedge d$ is prime, $deg(\mathbf{Z}_d) = t-1$ and a_{00} is the *secret* to reconstruct.

Next, the *dealer* identifies the most suitable t-dimensional subset of *players* to involve in the protocol by executing the proposed Quantum-Dijkstra Algorithm described in Sec. IV-A.

At this point, the dealer and selected players execute the proposed mutual authentication protocol described in Sec. IV-B

Once this authentication phase is correctly achieved, any single *player* is provided a *secret* key to communicate with the *dealer* (encrypting messages with a secure classical symmetric cipher as, e.g., AES-256). From now on, every single message exchanged between *dealer* and *players* will always be encrypted with the corresponding *secret* key.

The *dealer* will then share with the i-th player P_i the polynomial $\mathbf{G}(x_i, y)$, with $i \in \{1, \ldots, t\}$ and $x_i \in \mathcal{Z}_d$ being a random coefficient. This polynomial is later needed to let the player reconstruct its *share's shadow* as mathematically defined in the following.

The *Entangled States Sharing* is done by the *dealer* by building a d-dimensional GHZ state (i.e., the local Hilbert space is isomorphic to $\mathbb{C}^d$) in the form:

$$|\phi\rangle = \frac{1}{\sqrt{d}} \sum_{i=1}^{d-1} |\nu\rangle_1 \otimes \cdots \otimes |\nu\rangle_{t-1} \tag{2}$$

The GHZ state $|\phi\rangle$ contains exactly t particles, one for each participating *player* and plays a pivotal role in the protocol by distributing entangled particles to participants. The *dealer* will indeed send the entangled particle $|\nu\rangle_i$ assigned to *player* $P_i, \forall\, i \in \{0, \ldots t-1\}$. However, the real particle $|\nu\rangle_i$ is hidden in a j-particles stream through a BB84-like protocol by

adding an arbitrary number $a \in \{1, \ldots, j-1\}$ of decoy particles $|\delta\rangle$. This mechanism avoids/decreases the probability that an attacker gathers any further information or invalidates this step of the protocol [29], [30]. Additionally, the arbitrary number of the decoy particles represents a tuning parameter to better balance efficiency (less decoy particles) and security (more decoy particles) at every execution.

Hence, the *dealer* accomplishes the following steps:

1) transmits the entangled particle $|\nu_i\rangle$ to *player* P_i, $\forall\, i \in \{0, \ldots t-1\}$, by generating $a \in \{1, \ldots, j-1\}$ random quantum decoy particles $|\delta\rangle$ and randomly polarizing them in basis: $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$.
2) prepares the particle stream by placing, in a random spot, the real GHZ state $|\nu\rangle_i$ and obtaining a stream like the following: $|\delta\rangle_0 |\delta\rangle_1 \ldots |\nu\rangle_i \ldots |\delta\rangle_{\mathrm{a}}$
3) builds a classical message that explains, for each particle, the measurement base and expected outcome, while also specifying where the entangled particle is placed to allow the *player* to store and send it without applying any measurement.
4) shares one by one the particles, by distinguishing two cases:
   a) the i-th player is a directly connected node, then the entangled particles are directly sent over the fiber connection;
   b) otherwise, the target player i is reached by the entangled particles through multiple *Entanglement Swaps* by a series of intermediate nodes, that are alerted singularly by the *dealer* before the exchange begins. Since the *dealer* is the only available entity with full knowledge of the network structure and the hops needed to reach a *player*, it will instruct each intermediate hop to which specific interface the received entangled particle has to be swapped.
5) securely generates the parameters $l_i = r_i \bmod d$, $\forall i \in \{0, 1, \ldots, t\}$, where r_i, $i \in \{0, 1, \ldots, t\}$, are the high-entropy random values generated by the dealer employing a Cryptographically Secure Random Number Generator (*CSPRNG*). This process is designed to guarantee that l_i, $i \in \{0, 1, \ldots, t\}$, are uniformly random and inherently unpredictable, a necessity for secure and robust quantum operations, enhanced by their secure generation that protects the protocol from potential exploitation by malicious actors. In scenarios where reproducibility is required, such as during debugging or analysis, the dealer can ensure consistency by using a securely derived seed.

At this point, each single *player* P_i:

1) applies the pre-shared measurements and verifies the expected match. The protocol also takes into account a slight probability of having one or more erroneous measurements due to swap actions and particle decoherence over the fiber channel. This value can be dynamically adapted to any increase of the swaps to balance among security and protocol usability.
2) calculates its *shares' shadows* S_i defined as the individual pieces of information that collectively allow the reconstruction of the original secret:

$$S_i = \mathbf{G}(x_i, 0) \prod_{j \neq i}^{t} \frac{x_j}{x_j - x_i} mod\, d \tag{3}$$

where x_i are coefficients assigned to players, and d is the prime field size used for modular arithmetic. This ensures that only authorized subsets of participants can reconstruct the secret while maintaining security against unauthorized access.
3) computes its $|l_i \oplus S_i\rangle$ by embedding its own S_i that contains the secret, in the previously received GHZ state $|\phi\rangle$. By taking into account all the different embedding contribution of each participating *player*, the aforementioned $|\phi\rangle$ state evolves into $|\phi'\rangle$, as detailed in [16]:

$$\begin{aligned} |\phi'\rangle = \frac{1}{\sqrt{d}} \sum_{\nu=0}^{d-1} d^{-\frac{t}{2}} \sum_{l_1, \ldots, l_t} \omega^{k(l_1, \ldots, l_t)} \\ \times (|l_1 \oplus S_1\rangle) \ldots (|l_t \oplus S_t\rangle) \end{aligned} \tag{4}$$

where $\omega = e^{2\pi i/d}$ is the root of unity. This transformation ensures the secure embedding of the secret into the shared entangled state for later reconstruction. Furthermore, this operation can be achieved by using a QFT (Quantum Fourier Transform) circuit while exploiting the *Generalized Pauli Operators* [28].
4) measures the new obtained entangled particle in the computational base $\{0, \ldots, d-1\}$ and obtains a measure $M_i = S_i \oplus l_i \, \forall i \in \{0, 1, \ldots, t\}$, ready to be shared with the other players.
5) exchanges the data and calculates the final value of the secret, as:

$$\begin{aligned} \sum_{i=0}^{t} M_i &= M_1 \oplus \cdots \oplus M_t \\ &= (S_1 \oplus l_1) \otimes \cdots \otimes (S_t \oplus l_t) \\ &= (S_1 \oplus \cdots \oplus S_t) \otimes (l_1 \oplus \cdots \oplus l_t) \\ &= \sum_{i=0}^{t} S_i = S \end{aligned} \tag{5}$$

where $\oplus$ is the XOR operation. This ensures that only an authorized subset of participants can reconstruct the secret.

The data exchange among the players can be done in two ways:

- **Dealer as Distributor:** In this scenario, each *player* sends its measured result back to the dealer, encrypted with the shared secret key. The dealer verifies correctness (detecting and halting if there is any indication of tampering) and then distributes all of the collected

measurements to every participating *player*, along with a hash of the final secret. This ensures that any cheating attempt is revealed to all parties;

- **Bulletin Board:** Alternatively, the dealer plays a passive role by simply collecting and publishing the measurements on a publicly readable bulletin board, without actively verifying their correctness. Once every player has contributed its result, they all retrieve the posted measurements and use these values, along with the shared hash, to locally reconstruct and validate the secret.

Finally, at the end of the protocol the hash of the secret (i.e., via SHA3) is provided to all the *players*, to let them to check the correctness of the execution.

A. QUANTUM DIJKSTRA ALGORITHM

Dijkstra is a well-known path finding algorithm, and with this paper we reformulated it into a quantum version. To provide a real enhancement to the classical Dijkstra algorithm we propose to use a quantum algorithm to speed-up the extraction of the best costing path, thus we introduce the Optimized Quantum Minimum Search Algorithm (OQMSA) [31], a new method for calculating the minimum value from a random vector.

The Quantum Dijkstra algorithm is presented in Algorithm 1, while a flowchart is shown in Fig.3 and the OQMSA is described in Algorithm 2.

The introduction of the OQMSA approach, based on an enhanced and more accurate version of Grover's algorithm [32], reduces the time-complexity to $O(|V| * \sqrt{|V|})$ versus the $O(|V|^2)$ of the conventional Dijkstra's algorithm [33]. However, this efficiency gain is balanced against a small probability of extracting the incorrect minimum, because unlike classical algorithms, OQMSA can extract the exact minimum from an unsorted array with a success rate of 98%.

This quantum Dijkstra algorithm uses specifically designed links' weights given by:

$$C_{v,u} = (\kappa/\alpha) + ((1-\kappa) * \beta)) \tag{6}$$

where v is a node in a Graph G and $u \in N$ N = Neighbors(v, G).

In eq. (6), $\alpha \in [0, 1]$ represents a measure of the *quality of entanglement*, i.e., it describes the probability of having a successful entanglement between the two ends of the connection. To let a lower entanglement swap success probability, i.e., $\alpha \to 0$, to increase the link's weight, the aforementioned formula will use the inverse of α, i.e., $\frac{1}{\alpha}$. In such a way, for $\alpha \to 0$ we have $C_{v,u} \to \infty$.Since qubits cannot be copied, due to the No-Cloning Theorem, quantum swaps will need to occur to let *players* to route entangled particles [6]. To model how α can be calculated it is possible to refer to [34], [35].

The parameter β in eq. (6) accounts for the main characteristics of the fiber channel, i.e., capacity, non-linear interference noise, polarization-mode dispersion, and polarization-dependent loss [36], and is calculated as shown in the pseudo-code snippet in Algorithm 3.

Finally, the proposed formula in (6) involves the additional variable κ to introduce another degree of flexibility. It will act as a weight to balance between a formula oriented on swap capabilities (i.e., Distributed Computing) over fiber's quality and vice-versa.

If $\kappa \to 0$, then the process does not care about swapping qubits. Otherwise, if $\kappa \to 1$, it is needed to pass through links that allow for the best swapping results, without taking strictly into account all fiber's characteristics.

Algorithm 1: Quantum-Dijkstra's Algorithm with Edge Cost

Input: Graph, Source
Output: dist, prev

$vertices \leftarrow$ dim(Graph.Vertices);
foreach $v \in Graph.Vertices$ **do**
  $dist[v] \leftarrow |1\rangle^{\otimes vertices}$;
  $prev[v] \leftarrow null$;
while $dim(Q) \neq |0\rangle$ **do**
  $u \leftarrow OQMSA(dist[u])$;
  $Q.pop(u)$;
  foreach $v \in Neighbors(Graph, u)$ **do**
    $C_{u,v} \leftarrow$ Graph.Edges(u, v)
    $|var\rangle \leftarrow dist[u] + C_{u,v}$ **if** $|var\rangle < dist[v]$ **then**
      $dist[v] \leftarrow |var\rangle$;
      $prev[v] \leftarrow u$;
return $dist, prev$;

Algorithm 2: OQMSA [31]

Input: D, d' (Database and a Random Item)
Output: d_{min}

for $i \leftarrow 0$ **to** $\lceil \log(N) \rceil$ **do**
  $t_{max} \leftarrow \left\lfloor \frac{\pi}{2} - \arcsin\left(\frac{1}{\sqrt{N}}\right) \right\rfloor / \arcsin\left(\frac{1}{\sqrt{N}}\right)$;
  $t \leftarrow 1$;
  $\lambda \leftarrow \frac{6}{5}$;
  $r \leftarrow \infty$;
if $M/N > \frac{1}{9}$ **then**
  $t' \leftarrow randint(0, \lceil t \rceil)$;
  $|\phi'\rangle \leftarrow Grover\_Long(|\phi\rangle, t')$;
  $t \leftarrow t * \lambda$;
if $M/N < \frac{1}{9}$ **then**
  $|\phi'\rangle \leftarrow Grover\_Long(|\phi\rangle, t_{max})$;
$r \leftarrow Measure(|\phi'\rangle)$;
if $r < d'$ **then**
  $d' \leftarrow r$;
  $i \leftarrow 0$;
return $d_{min} = d'$;

Algorithm 3: Channel Measurement and Lookup Table Matching

Input: Nodes u, v
Output: Final value β

$\beta \leftarrow 0$;
$\{P_1, P_2, \ldots, P_n\} \leftarrow List\,of\,channel\,parameters$
foreach *parameter* P_i *in* $\{P_1, P_2, \ldots, P_n\}$ **do**
  $value_i \leftarrow$ `MeasureChannel(`P_i, u, v`)`
  $lookup\_value \leftarrow$ `LookupTable(`P_i, $value_i$`)`
  $\beta \leftarrow \beta + lookup\_value$
return β;

Equation (6) will then be employed to calculate each path cost, inside the proposed network, and build a *Dijkstra* path-matrix which will allow the *dealer* to identify the most suitable *players* to involve in the protocol execution. Path cost computation can be achieved in two different ways, both of them are equally feasible inside this protocol and the protocol's settings will specify which one to use:

1) *Single Source Computation*: only the source node initiates the calculation of route costs. It iteratively updates the cost to reach each node based on the distances to neighbors. Once a node's minimum cost is determined, it is considered "visited," and the algorithm continues to the next unvisited node with the smallest cost.
2) *Distributed Variants*: In some distributed implementations every node may independently compute the cost to reach its neighbors. This can be useful in dynamic networks where topology changes frequently. Each node can then share its cost information with its neighbors, allowing for a more collaborative approach to computing paths but require additional *player*-to-*player* communications and hardware for *players* quantum devices.

Algorithm 4: Weight Calculation Using Swap Success and Lookup Parameters

Input: v, u, κ
Output: $C_{v,u}$

$\alpha \leftarrow$ `EstimateSwapSuccess(`v, u`)` ;
$\beta \leftarrow$ `ParametersEstimation(`v, u`)` ;
$C_{v,u} \leftarrow \left(\frac{\kappa}{\alpha}\right) + ((1-\kappa) * \beta)$;
return $C_{v,u}$;

The primary advantage of such a dynamical weighting system lies in its ability to dynamically adjust the selection of participants based on real-time network conditions, such as quantum noise levels, distance between nodes, and the reliability of individual quantum channels. This flexibility enables the system to prioritize more stable or efficient routes, ensuring that the quantum keys are securely and efficiently distributed, even in fluctuating network environments.

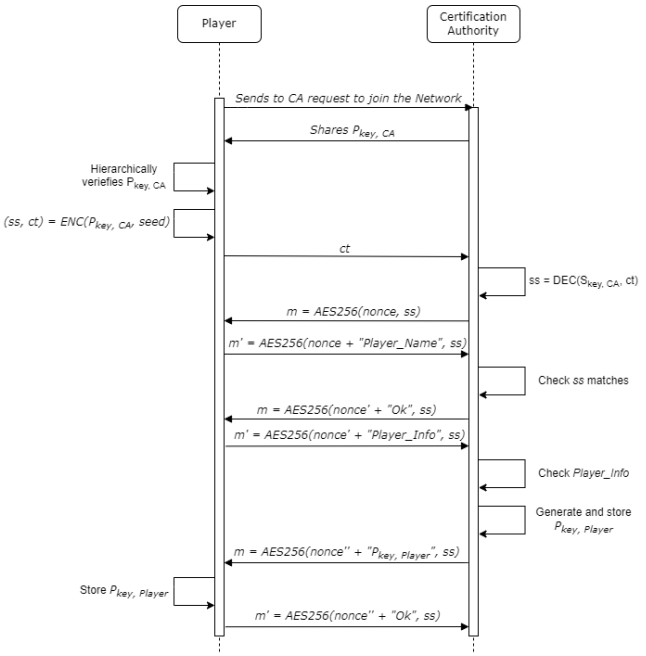

FIGURE 2: How a new Player joins the protocol.

An example of how the *Decoding Error Probability*, over a quantum channel Λ, impacts over the computation of the β parameter is provided in Table 2. Basing on the error level, β will be added with corresponding *score*.

Table 2 takes into account also for the *Quantum Information sent per qubit*, i.e., $\frac{1}{N}Q^{\epsilon}(\Lambda)$, which expresses the rate at which quantum information can be sent reliably through a quantum channel Λ when the number of uses of the channel grows large, corresponding to a specific *Decoding Error Probability* ϵ.

Once the weighting system is applied to the entire network, a situation like the one proposed in Fig. 4 is gathered. At this point, the most suitable t *players* (4 in this example) are chosen to be participating to the next protocol iteration.

B. AUTHENTICATION

For a new player to be admitted into the network, it must provide its *Public Kyber Key* to the *Certification Authority* (the *dealer* in this scenario). The *CA* will then register the new player. This process has been detailed in Fig. 2.

1) A *player* wants to join the protocol network and, to be accepted, it must at first register its public key with a *Certification Authority*. By taking advantage of *CRYSTALS-Kyber* primitives [37], *CA* shares its *Public*

N	Decoding Error probability (ϵ)	$\frac{1}{N}Q^{\epsilon}(\Lambda)$	Score
10^5	10^{-2}	≈ 0.320	3
10^5	10^{-6}	≈ 0.330	4
10^5	10^{-10}	≈ 0.360	5

TABLE 2: **Quantum information sent per qubit ($\frac{1}{N}Q^{\epsilon}(\Lambda)$)**

Key with the recipient;

2) A *participant* uses a hierarchically superior *CA* to assess $P_{key,CA}$ authenticity;
3) *Participant* computes *ciphertext* (ct) and a *secret-shard* ss by employing $P_{key,CA}$ and a random *seed*. *ct* is sent back to *CA* to let it to compute ss as well;
4) To confirm that both parties agreed on the shared key, and to avoid any possible reply attack, *Certification Authority* uses a symmetric encryption algorithm (i.e., AES-256) to encrypt a *nonce* and send it to the *Player*. If the latter has correctly executed the calculations, it will be able to decrypt the *nonce* and send it back by attaching to it its name;
5) *CA* checks the ss was correctly shared between them and sends back an "Ok" message to the recipient, meaning it is waiting for its information to begin with the key creation. The *Player* does exactly what the *Certification Authority* was waiting for and shares its personal information;
6) *CA* validates data and creates a new entry for this new acquired participant. Then, it randomly generates $P_{key,Player}$ and shares this key with the recipient. The *Player*'s Public key is combined with a nonce and shared and an encrypted message. By achieving so, an eavesdropper is not able to fingerprint the key, avoiding to leaking information on who might get involved in the protocol.

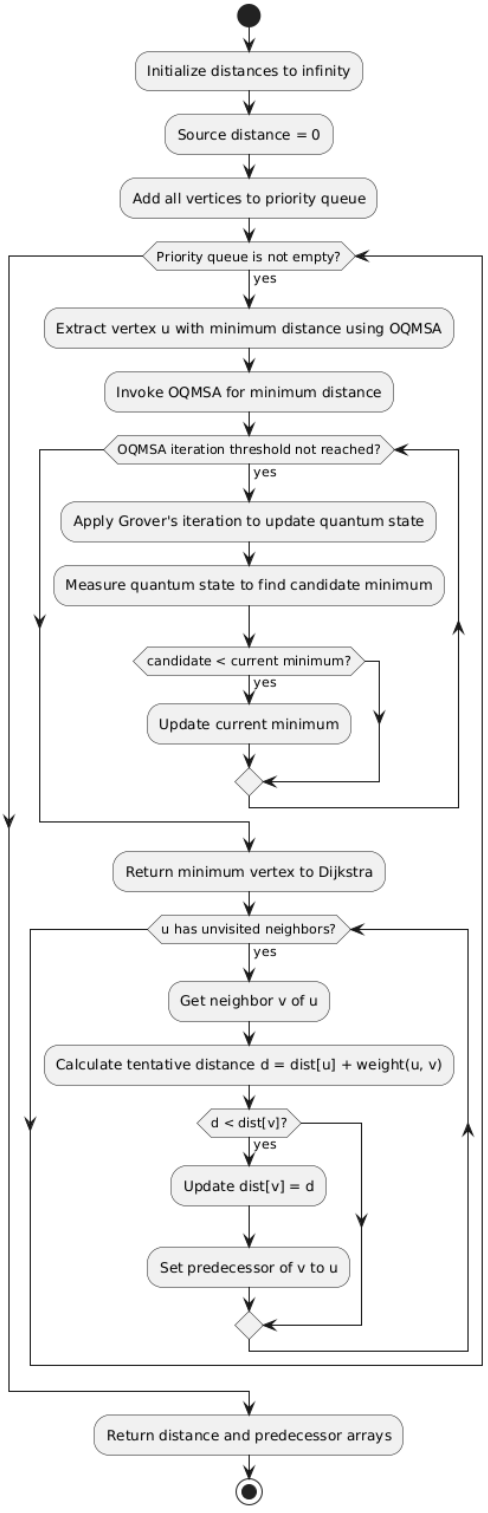

FIGURE 3: Flowchart detailing Quantum Dijkstra execution.

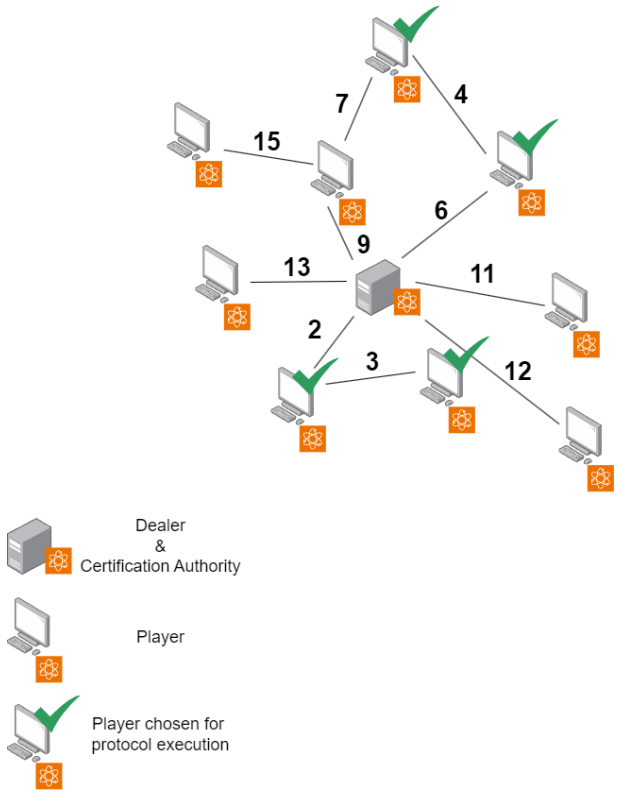

FIGURE 4: Distributed Network Topology with weighted links.

V. VALIDATION OF THE PROPOSED QSS PROTOCOL

In this section, we comprehensively validate our QSS protocol by examining both theoretical and practical aspects of

its performance. First, we provide a numerical example to illustrate how each step of the protocol unfolds in a (t, n)-threshold setting, demonstrating how the secret is securely shared and accurately reconstructed using modular arithmetic and entangled states. We then describe a Qiskit-based implementation, showing how our theoretical framework can be mapped onto qubit-based circuits using approximate amplitude embeddings and quantum gate sequences. This simulation explores realistic effects such as noise, multi-hop entanglement swaps, and decoy qubits, highlighting how the protocol can be adapted for current and near-term quantum devices. Finally, we analyze the protocol's fairness, discuss its compliance with the extended *CIA Triad* (Confidentiality, Integrity, Availability, Authenticity, Accountability, and Non-repudiation), and evaluate its resilience against potential attacks, confirming the overall security and robustness of the proposed scheme.

A. A NUMERICAL EXAMPLE

To improve clarity, a numerical example of the proposed QSS protocol is provided to implement a $(t, n) = (3, 5)$ threshold scheme, so that any 3 of the 5 players can collaborate to reconstruct a secret S with a prime field size for modular arithmetic $d = 7$ (which was deliberately selected to facilitate a clearer understanding of the subsequent calculations).

Let P_i, $i = 1, \ldots, n$, be the $n = 5$ players and D be the dealer. The network topology chosen for this example is shown in Fig. 5. In the following, we go through every phase of the algorithm, from participant authentication to secret reconstruction.

- **Authentication** Before any secret distribution or quantum communication begins, mutual authentication is performed using CRYSTALS-Kyber, so that the dealer knows it is interacting solely with legitimate players, and each player is assured of the dealer's authenticity. The steps are as follows:
  1) Each player registers its public key with the dealer (acting as the CA).
  2) The dealer, using Kyber, securely establishes shared symmetric keys ss_i with each player P_i, which are used to encrypt all classical communications.
- **Secret-Sharing Polynomial** The dealer chooses the secret $S = a_{00} = 3$. Arithmetic takes place in $\mathbb{Z}_7$. The dealer constructs a symmetric polynomial as defined in eq. (1), by encoding the secret and using random coefficients:

$$\mathbf{G}(x, y) = 3 + 2x + x^2 + 4y + 5xy + 6x^2y + y^2 + 3xy^2 + 2x^2y^2 \bmod 7. \quad (7)$$

- **Player Selection by Quantum-Dijkstra** The dealer applies the Quantum-Dijkstra algorithm to select the subset of $t = 3$ players that will participate. Assume the following link costs, obtained by eq. (6) with $\kappa = 0.5$ and specific entanglement success probability α_i and channel parameter β_i, per each player P_i, for $i = 1, \ldots, n$:
  - $D \to P_1$: $\alpha_1 = 0.9$, $\beta_1 = 2$, $C_{D,P_1} = 1.56$.
  - $P_1 \to P_2$: $\alpha_2 = 0.8$, $\beta_2 = 1.5$, $C_{P_1,P_2} = 1.375$.
  - $P_1 \to P_3$: $\alpha_3 = 0.85$, $\beta_3 = 1$, $C_{P_1,P_3} = 1.09$.
  - $P_2 \to P_4$: $\alpha_4 = 0.7$, $\beta_4 = 2$, $C_{P_2,P_4} = 1.71$.
  - $P_3 \to P_5$: $\alpha_5 = 0.9$, $\beta_5 = 1$, $C_{P_3,P_5} = 1.06$.

  From the dealer's perspective:

$$D \to P_1 = 1.56, \quad D \to P_3 = 2.65, \quad D \to P_2 = 2.935. \quad (8)$$

  Paths to P_4 and P_5 are even costlier. Thus, the dealer selects P_1, P_3, and P_2 as participants, as shown in Fig. 5.

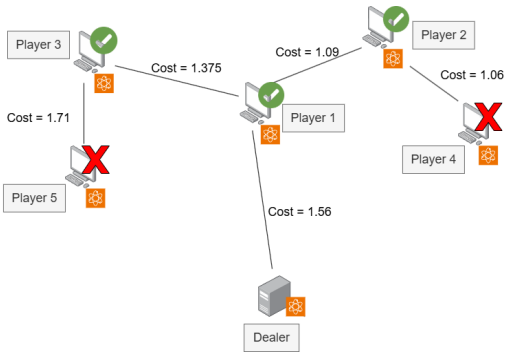

FIGURE 5: Network topology for the numerical example.

- **Distributing Shares and Polynomials** Assign $x_1 = 1$, $x_2 = 2$, $x_3 = 3$, as depicted in eq. 3, and evaluate $\mathbf{G}(x_i, 0)$:

$$\begin{aligned} \mathbf{G}(1,0) &= 3 + 2(1) + (1)^2 = 6 \bmod 7 \\ \mathbf{G}(2,0) &= 3 + 2(2) + (2)^2 = 11 \equiv 4 \bmod 7 \\ \mathbf{G}(3,0) &= 3 + 2(3) + (3)^2 = 18 \equiv 4 \bmod 7 \end{aligned} \quad (9)$$

- **Entanglement Distribution**
  The dealer prepares a 7-dimensional GHZ state:

$$|\phi\rangle = \frac{1}{\sqrt{7}} \sum_{\nu=0}^{6} |\nu\rangle_{P_1} |\nu\rangle_{P_2} |\nu\rangle_{P_3}. \quad (10)$$

  To send each player's qudit securely, the dealer uses decoy particles. For example, to P_1 it sends:

$$|\delta_0\rangle, |\nu_{P_1}\rangle, |\delta_1\rangle \quad (11)$$

  with expected outcomes for $|\delta_0\rangle$ and $|\delta_1\rangle$ provided via encrypted classical channels. P_1 measures these decoys to confirm no eavesdropping. After verification, P_1 stores $|\nu_{P_1}\rangle$. A similar procedure is followed for P_2 and P_3, using entanglement swaps if intermediate hops are needed. Each player thus securely obtains their share of the GHZ state.

- **Share Shadows' Computation** By eq. (3), we get:

$$S_i = \mathbf{G}(x_i, 0) \prod_{j \neq i} \frac{x_j}{x_j - x_i} \mod 7. \tag{12}$$

For P_1, we have $S_1 = 6 \cdot 2 \cdot \frac{3}{2}$.
Since in $\mathbb{Z}_7$ $2^{-1} = 4$ and $\frac{3}{2} = 3 \cdot 4 = 12 \equiv 5 \mod 7$, it results:

$$S_1 = 6 \cdot 2 \cdot 5 = 60 \equiv 4 \mod 7. \tag{13}$$

For P_2: $S_2 = 4 \cdot \frac{1}{6} \cdot 3$, with $6^{-1} = 6 \mod 7$, hence:

$$S_2 = 4 \cdot 6 \cdot 3 = 72 \equiv 2 \mod 7. \tag{14}$$

For P_3: $S_3 = 4 \cdot \frac{1}{5} \cdot \frac{2}{6}$, with $5^{-1} = 3 \mod 7$ and $6^{-1} = 6 \mod 7$, thus:

$$S_3 = 4 \cdot 3 \cdot (2 \cdot 6) = 144 \equiv 4 \mod 7. \tag{15}$$

Finally, the secret is given by:

$$S_1 + S_2 + S_3 = 10 \equiv 3 \mod 7. \tag{16}$$

- **Embedding S_i and Dealer's Random Values l_i** Before the final measurement, the dealer assigns random l_i to the three players, e.g., $l_1 = 1,\ l_2 = 2,\ l_3 = 0$ and send them via their secure encrypted classical channel to ensure the measured result $M_i = S_i + l_i (\mathrm{mod}\, 7)$. After performing the appropriate generalized Pauli operators and QFT, each player measures their qudit, i.e., $M_1 = S_1 + l_1 = 4 + 1 = 5$, $M_2 = S_2 + l_2 = 2 + 2 = 4$, and $M_3 = S_3 + l_3 = 4 + 0 = 4$ and send them to the dealer (or publish them on a bulletin board).
- **Secret's Reconstruction** Knowing all l_i and M_i, the dealer reconstructs the secret as

$$\begin{aligned} &(M_1 + M_2 + M_3) - (l_1 + l_2 + l_3) = \\ &= 6 - 3 = 3 \mod 7, \end{aligned} \tag{17}$$

which matches $S = 3$.
- **Verification and Fairness** The dealer provides a hash $H(S)$ to all players. If any player had cheated by altering M_i, the final result would fail the hash check. Thus, the protocol ensures fairness: all players either reconstruct the correct secret or detect malicious behavior.

This numerical example demonstrates that the theoretical concepts behind the mathematical formulation of this protocol are valid and allow to securely reconstruct the *shared secret*. Although the focus is mainly on the computational components and all quantum components are approximated, the provided QSS protocol cannot be assimilated to a mere variant of a classical one. In fact, it offers a superior level of security compared to classical secret-sharing methods by leveraging the foundational principles of quantum mechanics, such as the no-cloning theorem and quantum entanglement. Any interference with the quantum states is immediately detectable, as it disturbs the entire system and alters the outcomes of quantum measurements, hence, the mere act of observation by an eavesdropper collapses the quantum state, introducing detectable anomalies.

B. QISKIT SIMULATION

For the validation of the proposed QSS protocol in realistic case studies, we selected the Qiskit framework [38] to compare the protocol's performance in either a locally simulated environment and within the IBM Quantum Network.

Qiskit offers a robust and widely adopted framework for qubit-based quantum computing, making it a suitable platform for prototyping the proposed QSS scheme. Despite the protocol's theoretical reliance on higher-dimensional (qudit) states, Qiskit's modular structure and extensive libraries facilitate the necessary abstractions, such as amplitude embeddings and partial-phase rotations, to approximate d-dimensional operations. Furthermore, Qiskit integrates classical control-flow features, enabling the simulation of multi-hop entanglement swaps and conditional measurements, both of which are central to the multi-party routing paradigm in our protocol. The availability of a large community, coupled with its open-source nature, also ensures continuous improvement of Qiskit's toolset, thereby making it an accessible and effective environment to conduct proof-of-concept demonstrations of our QSS implementation.

As described in the following, we developed our Qiskit-based code overcoming the challenge stemming from overhead and multi-level mismatch, thus demonstrating that real or near-term hardware might partially realize a multi-hop secret sharing protocol. It stands as a proof-of-concept that code-level modifications, such as, e.g., embedding truncated amplitude states and partial-phase gates, layering decoy qubits for intrusion detection, conditionally simulating entanglement swaps, can bring the theoretical protocol to life under substantial engineering constraints. Obviously, future expansions on hardware, with direct qudit support or better mid-circuit measurement handling, would reduce much of this overhead. For this simulation, we assume the prime field size for modular arithmetic to be $d = 3$, to reduce the overhead required to emulate the corresponding states, i.e., $\{0, 1, 2\}$, on qubit-based hardware and to cope with the constraints in the *Qiskit* simulation environment. This choice stems from the pragmatic difficulties of bridging multi-level cryptographic protocols and hardware built around binary qubit logic. While the choice of $\mathrm{mod}3$ is not the ideal representation for a higher-dimensional qudit space (e.g., $\mathrm{mod}7$ or above), it allows us to substantially reduce the complexity of embedding multi-level quantum states within *Qiskit*'s qubit-based framework. Specifically, higher moduli require more intricate amplitude initializations and extended gate sequences to accurately simulate the necessary phase rotations, which can rapidly increase circuit depth and susceptibility to noise. By restricting our attention to a three-element field, we retain the essential structure of threshold-based secret sharing, namely that each player's measurement contributes one piece of the final secret—while avoiding the steep overhead that full-fledged qudit simulation would impose on contemporary hardware and software. Hence, although it does not perfectly capture the protocol's higher-dimensional capability, using $\mathrm{mod}3$ still provides a clear

and effective demonstration of our scheme's core feasibility and design principles.

Once qubit machines or more refined multi-qubit entanglement topologies will become routine, higher dimensions could be considered with few design compromises. In the interim, focusing on $\mathrm{mod}\,3$ ensures we still convey the key features of threshold-based secret reconstruction and partial-phase embeddings, while containing the mismatch between the theoretical arithmetic and the classical-quantum hybrid code Qiskit can realistically support. Three states can be approximated by either an eight-dimensional (three-qubit) space or a single qubit with amplitude $\sqrt{\frac{2}{3}}$ in $|0\rangle$ and $\sqrt{\frac{1}{3}}$ in $|1\rangle$ as a conceptual demonstration, while the partial-phase rotations become $\frac{2\pi}{3}\times(S_i+\ell_i)$ in $\mathrm{mod}\ 3$. For further details about the rationale for this choice and the implementation constraints we refer the interested reader to the related *GitHub* repository [39].

We represent the network topology in a JSON-based adjacency structure (e.g., `"Player 1" : {"Player2" : {"alpha": 0.7, "beta": 2}}`). The system then identifies the players most suitable to receive the partial shares of the secret by employing eq. (6) to calculate each link's weight while the *Quantum-Dijkstra* algorithm in **Algorithm** 1 selects minimal elements among unvisited vertices. (in Fig. 6 an example of *OQMSA* quantum circuit implementation is shown).

This implementation aligns with the theoretical requirement that node selection should leverage quantum speedups, but remains subject to *Qiskit*'s practical constraint of encoding distances in a discrete, power-of-two format. Hence, weights are encoded as power-of-two values instead of generic floating points values. After the algorithm determines which node has the lowest cost, it updates the edges associated with that node, thereby following the iterative steps of the standard Dijkstra procedure, allowing the code to flawlessly provide a correct result even tough an hidden conversion is taken.

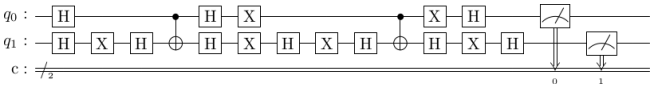

FIGURE 6: An example of *OQMSA* implementation to find minimum value in the distance set.

Our code then executes a cryptographic handshake between the dealer and each chosen player. This handshake uses post-quantum primitives (*Kyber*) to generate ephemeral shared keys, which are then used in classical AES encryption for subsequent communications. The handshake implements a short sequence of messages in which each party verifies a nonce and finalizes an ephemeral key. We therefore satisfy the protocol requirement that each participant be uniquely authenticated and that no eavesdropper can intercept or spoof their partial share transmissions.

The secret sharing is implemented by building separate three-qubit circuits per player, where a qubit is used to approximate the $(S_i + l_i)$ embedding via partial-phase gates, and two decoy qubits to detect tampering or to stand in for an entanglement-swapping process for multi-hop routes. In a perfect multi-level system, each $\mathrm{mod}d$ step could be a single operation, but on a qubit device we rely on rotating one qubit by an angle proportional to $\frac{2\pi}{d}\times(S_i+l_i)$. We similarly embed truncated amplitude initializations to approximate multi-level states: in practice, this places the zero state and the one state in a ratio selected to mimic the $\sqrt{\frac{2}{3}}$ vs. $\sqrt{\frac{1}{3}}$ amplitude distribution for $\mathrm{mod}\,3$ demonstrations. Each time a qubit is measured, the ephemeral quantum information is effectively destroyed, so real devices would require repeated circuit runs to gather statistics.

An interesting component of our protocol simulation lies in demonstrating how multi-hop entanglement swaps can be chained together to pass a quantum state across multiple intermediate nodes. Conceptually, an entanglement swap is performed by entangling two pairs of qubits and then carrying out a *Bell-state measurement* (*BSM*) on one qubit from each pair, thereby transferring or redirecting the entanglement from one node to another without requiring a direct physical link. In qudit-based hardware or ideal multi-level systems, this routine can be carried out in a single continuous sequence of gates and measurements. However, within the constraints of *Qiskit*'s qubit-based environment and typical two-level hardware assumptions, we approximate the logic by introducing new qubit lines whenever a multi-hop route is needed.

The code for the function `simulateEntanglementSwap` begins by identifying how many newly introduced qubits (or lines) are required. If the circuit already has three qubits (q_0 for data, q_1 and q_2 as decoys), and we desire n intermediate nodes, we then create a fresh circuit with $3+n$ qubits. In this new circuit layout, the original data qubit (old q_0) is placed at index 0, while the original decoys occupy the last two qubit indices of the new circuit. Each intermediate node is allocated exactly one extra line in the new circuit.

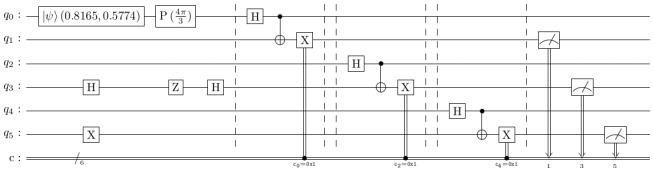

FIGURE 7: Implementing an entanglement swap to simulate an intermediate *player* routing data to another player.

Within this new circuit, each entanglement swap is realized through a short sequence. First, a *Hadamard* is applied to the current source qubit, placing it in a superposition, followed by the execution of a *controlled-NOT* onto the newly introduced qubit (the target line). This pair of operations mimics the creation of a *EPR* (*Einstein-Podolsky-Rosen*) pair when combined with a classical detection routine. Because the protocol aims to ensure that the final entanglement is carried forward, we measure the source qubit on the computational

basis and conditionally apply an X operation (i.e., a NOT-gate) on the target if the result is 1. This step effectively disentangles the source qubit while transferring the relevant quantum correlations to the newly allocated line (an example is given in Fig. 8).

If multiple intermediate nodes are present (say $n = 2$), the procedure is repeated chain-like: the original data qubit (index 0) is connected to a newly introduced line (index 1), then line (index 1) is itself connected to line (index 2), concluding with a X correction (i.e., applying a NOT-gate) if required, as shown in Fig. 7. At the end of these chained swaps, the final data line stands at index n, meaning the entire data state (initially carried by old qubit 0) is now localized in qubit n. The decoy lines, for simplicity sake, remain placed at the last two indices and will be measured independently to confirm no tampering or to detect intrusion attempts. The final measurement of the data is thus carried out on qubit n, ensuring the ephemeral path was effectively "swapped" across n intermediate steps.

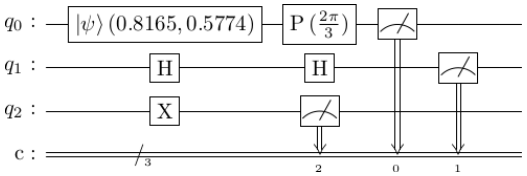

FIGURE 8: An example of *GHZ* and decoy states shared by the *dealer* to a participating *player*.

Once a player's partial circuit is measured, the measured result is encrypted with AES and returned to the dealer. The dealer accumulates all these partial results in a dictionary keyed by the player object. Since our code works modulo d, each partial share $(S_i + l_i)$ is effectively one classical bit in the simplified example, though in principle a multi-level measurement would yield an integer in the range $\{0, 1, \ldots, d-1\}$. The code then reassembles the final secret by summing the bits from all players and subtracting the known offsets ℓ_i. If the resulting value matches the original secret, the simulation concludes that the QSS subroutine has succeeded.

Finallly, the summation in modulo d is realized by reconstructing the values from the measured results and subtracting out the known offsets. This upholds the threshold logic introduced in the paper, in which the dealer, upon collecting all partial bits, recovers the global secret only if the correct subset of participants cooperates.

Despite the applied approximations, the simulator represents a good framework to stress-test the feasibility of the logic under the scarcities of qubit-based devices, being each step carefully designed to reflect the theoretical essence of multi-level threshold secret sharing, complete with amplitude-based embeddings and ephemeral qubit swaps. In effect, it emulates qudit transitions, advanced feed-forward, and multi-hop entanglement, yet does so with the overhead of extra gate sequences and ephemeral qubit lines. This approach highlights both the ingenuity needed to reconcile the protocol's theoretical demands with existing two-level technology, and the substantial leap required before a real quantum machine can natively support such a multi-dimensional, multi-hop secret sharing routine in a single hardware pass. Thus, the simulator stands as a valuable demonstration of the design's implementability, although it remains an approximation of features still unimplemented or partially supported in available quantum hardware and Qiskit itself.

With reference to Fig. 11, simulations achieved on a local *Qiskit* instance show how, for directed connected instances, as shown in Fig. 9, and without any Quantum Error Correction (QEC) mechanism, quantum measurements are able to return a correct result in the 50% of runs. On the other hand, for the ones involving one entanglement swaps, as shown in Fig. 10, this drops to around the 42%. This results confirm the feasibility of the proposed QSS protocol, since any real implementation usually relies on QEC mechanisms to improve efficiency, thus significantly increasing the shown success rate.

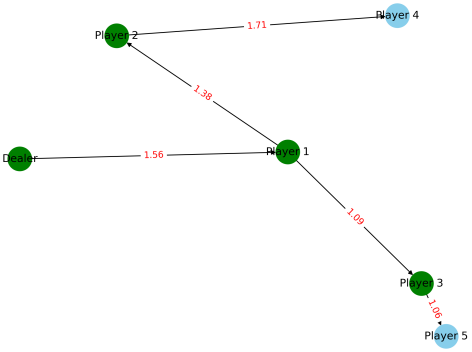

FIGURE 9: Example Graph of the set of *Players* chosen for a simulation round.

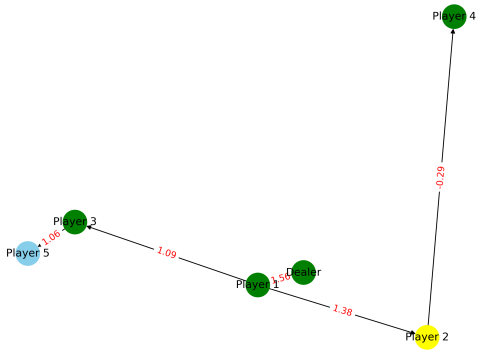

FIGURE 10: Simulating entanglement swap by forcing a non-dealer's connected *player* to join (Yellow node is an intermediate non-participating *player*)

The same circuits implemented for the previously men-

tioned local *Qiskit* simulations were also tested on the IBM Quantum Network, trough *Qiksit APIs*, which yielded closely aligned outcomes, albeit with slightly reduced success rates (approximately 45% in the absence of swaps, and roughly 40% with one swap), as shown in Fig. 12. This minor drop is expected, as the IBM network's noise model reflects hardware-level fluctuations more precisely than a simplified local simulator. Nonetheless, given the numerous adaptations and subtleties necessary to emulate a *qudit*-based protocol on purely *qubit* hardware, these experiments substantiate the feasibility of implementing the proposed scheme on real devices.

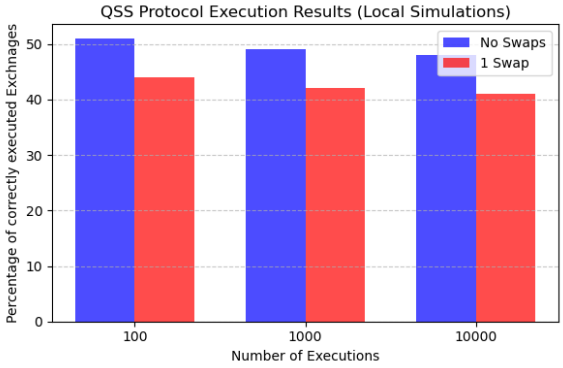

FIGURE 11: Summarized results of Qiskit Protocol Executions (local simulations).

The results also underscore the need for robust QEC, since qubits exhibit acute sensitivity to environmental noise. Overall, the simulations confirm that the protocol achieves its intended functionality, providing a secure threshold mechanism for distributing quantum shares and ensuring that only authorized sets of participants can reconstruct the secret in a privacy enhanced environment with the usage of a Quantum-Dijkstra Algorithm to select the most suitable players.

FIGURE 12: Summarized results of Qiskit Protocol Executions (IBM Quantum Network simulations)

C. FAIRNESS

In the proposed protocol, this constraint is achieved by relying on the followings:

1) In the first protocol implementation, the *dealer* acts as a *broker* between all the parties. Indeed, it gets the measurements from them and, as first, checks for any forging attempts. Hence, all the *players* are able to both gather the final result or know that someone tried to hijack it. For a cheater, it is difficult to correctly achieve its intent, since the *dealer* knows which share has been given to it and which measurement is expected to be shared. In this case the role of this broker is dominant, but at the cost of a trusted third-party, not always available, there is a hidden security guarantee that can not be ignored;
2) In the second implementation, which exploits the usage of a *Bulletin Board*, the *dealer* has a passive role, since it just acquires the measurements and then post them over a *bulletin board*. Since the broker does not actively verify the correctness of the protocol, there could be a non-zero chance of the cheater being able to let all the fair players to gather wrong results.
Dealer's final result hash is shared at the end not as a mere case, but it is chosen to avoid any possible brute-force attempts or to give any possible insight to the malicious *player*. Nonetheless, the attacker(s) could still be able to generate an hash collision by providing a fake measurements:
By considering how our proposed protocol works, and with the hypothesis of having n total *players*, 1 *dealer*, t actively involved *players*, let's assume there are f fraudulent *players*, with $t - f$ non-cheating *players*. Cheaters could provide $M'_j, j \in \{0, \ldots, f-1\}$ fake measurements with the objective of obtaining an hash collision, as shown in Equation 18:.

$$\begin{aligned} &\sum_{i=0}^{t} M_i = S \\ \Rightarrow &\sum_{i=0}^{t-1-f} M_i + \sum_{j=0}^{f-1} M'_j = S' \\ &H(S) = H(S') \end{aligned} \tag{18}$$

where f is the number of fraudulent players, M'_j are their fake measurements, and H is the cryptographic hash function. This analysis highlights the robustness of the protocol against collusion. At this point, if cheaters are able to correctly counterfeit the hash calculation, they will be the only one gathering the original *secret*, while the remaining host parties will be provided with a fraudulent one.

D. CIA TRIAD (EXTENDED)

The following paragraphs elucidates how each attribute of the Extended CIA Triad is demonstrated within the protocol, highlighting its comprehensive approach to safeguarding quantum communications.

Confidentiality The adoption of the CRYSTALS-Kyber Key Encapsulation Mechanism (KEM) allows both the Dealer

and players to mutually authenticate each other and securely establish a shared encryption key. This key, generated during the registration phase, serves as a unique identifier for each participant, thereby enhancing the security of the system. Once established, this shared key is used to ensure data confidentiality through a symmetric encryption scheme, such as AES-256. Since Kyber is a post-quantum algorithm designed for classical environments, its integration with symmetric encryption like AES-256 is more straightforward compared to quantum-based cryptographic schemes, which typically require complex setups and may be more vulnerable to noise and implementation challenges.

A different analysis needs to be made for GHZ states. Even if an attacker is able to capture and read its value (destroying the particle) it will not gather any useful information, since the secret calculation is made by the *player* once it received the entangled particle. Hence, any manipulation can only cause a delay in the protocol's execution; All the classical information exchanged between the parties can rely over a, e.g., generic TCP-fashion network communication where each datagram is equipped with a CRC-like code to allow for integrity checks.

Instead, while dealing with quantum particles exchange, a different mechanism needs to be applied to improve the integrity of the shared entangled state. To do so, in the proposed scheme the *dealer* adds *decoy particles* while sharing GHZ states with *players*. This allows to estimate the likelihood of an attacker to hijack the protocol. The more particles are incorrectly received by the *player*, the more probabilities there are that the communication is under attack. If too many particles do not align with the expected outcome, the entire process is repeated, allowing for managing the integrity of the delivered $|\nu\rangle_i$ state.

Availability. In classical network communication, information could, when needed, re-transmitted. Quantum particle distribution, on the other hand, is more susceptible to volumetric attacks (e.g., Denial-of-Service attacks). However, if at least one of the *players* has not received its the entangled particle the *dealer* will ask every *player* to stop and restart the *Entangled States Sharing* phase. The protocol performance will be degraded but it would be still operative and able to be correctly completed;

Authenticity. As per the previously mentioned authentication mechanism, based on a *Certification Authority* and the *CRYSTALS-Kyber* KEM, each *player* authenticates the *dealer* and the latter authenticates all the involved *players*;

Accountability. Since each message signed with a specific *secret share*, both *dealer* and *players* are able to confirm their identities, as per the usage of *CRYSTALS-Kyber*. To also take into account for attacks where a *player* is stolen of its keys, *network probes* can be placed in the network intercepting traffic and allowing for further review of the packets' source, allowing for any rogue node to be identified;

Non-repudiation. Once a message has been signed with a specific *secret share*, it is not possible for a *player* or even the *dealer* to repudiate it. Since the *ss* is owned by them (generated with the aid of a pair of public/private keys), no other can use it to sign messages (if no attacks involving key-starling activities are achieved).

Security Analysis. This section aims to provide a detailed examination of the protocol's security features, offering insights into its strengths:

- *Denial of Service*: in the proposed scheme, an attacker may attempt to disrupt either quantum or classical communications, or both. The quantum network under consideration functions as a distributed system, with multiple hosts not necessarily directly connected to the *dealer*. This design introduces an additional layer of redundancy; even if an attacker targets one or more classical or quantum channels, the *dealer* can respond by removing *players* under DoS attack, in favor of incorporating new ones. Moreover, the network topology is not publicly disclosed, and each player is aware only of their immediate neighbors. Consequently, rogue entities can only impact a limited portion of the network. Additionally, as discussed in relation to the protocol's *accountability* feature, strategic placement of *network probes* allows for traffic monitoring and the potential to *mute* non-collaborating nodes, thereby maintaining network integrity and security.

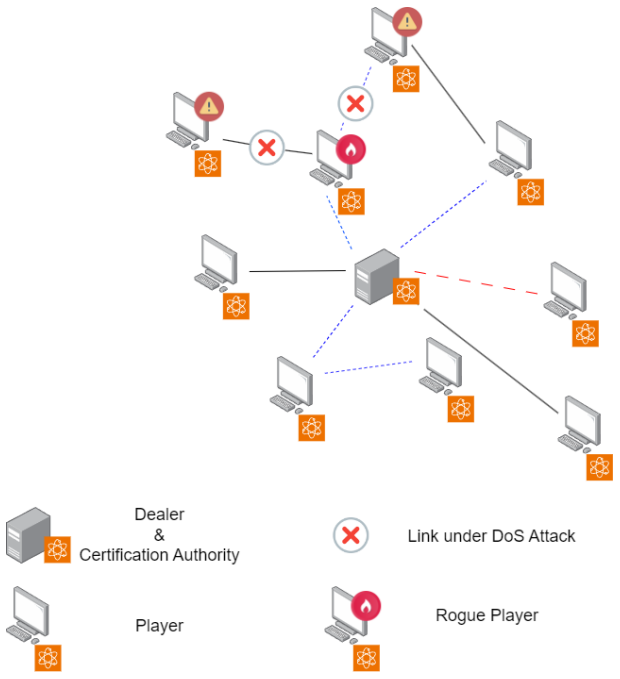

FIGURE 13: How a DoS can impact proposed network.

As illustrated previously, *players* participating in a protocol round are indicated with a blue dotted link. A rogue *player* may attempt a Denial of Service (DoS) attack on nearby connections, obstructing the *dealer* from reaching them, as shown in Fig. 13. However,

if a player is compromised, the protocol adapts by selecting another player based on the *Quantum-Dijkstra* algorithm, indicated by a red dotted link. This approach ensures the protocol's resilience and continuity despite potential disruptions. Generally, quantum communications are inherently more susceptible to interference. An attacker could attempt to delay the protocol by estimating which of the v quantum-exchanged particles contains the GHZ state. By identifying and replacing the genuine particle with a fraudulent one, a rogue node can disrupt the protocol, causing it to fail due to the loss of entanglement necessary for embedding the *secret shards* within a random photon. Probabilistically, an attacker altering a single particle has, at most, a $\frac{1}{v}$ chance of successfully targeting the entangled particle without affecting the control particles. Moreover, the attacker might estimate a higher likelihood of the GHZ state being placed towards the end of the stream due to the photon's unstable nature and susceptibility to rapid decoherence. Thus, as v increases, so does the likelihood of the entangled state being located in the latter part of the sequence. To mitigate this, v should be carefully chosen to securely obscure the entangled particle while not being too large to avoid facilitating tailored guessing by the attacker.

- *Reply attack*: To prevent a malicious entity from reusing pre-exchanged information to deceive another party several countermeasures are employed. During the registration of a new *player* and protocol's AUTHENTICATION phase, *nonces* are used. Consequently, an attacker cannot rely on previously shared messages, as each message is based on a*nonce* and a different *shared secret* (ss), which is re-generated for each protocol run. Additionally, private keys are never publicly shared, ensuring that no attacker can recover them without access to a participant's device. Therefore, an attacker has only two potential methods to introduce themselves into the protocol:
  - *Entangle shared particles*: *Dealer* sends a *player* v photons. Each of them can be intercepted by an attacker, locally entangled, and then let to flow to the recipient. In this way, the rogue node disposes of something it can use to try to gather further information. Anyway, since the encoding information and the GHZ state position are shared as an encrypted message, this fraudulent actor does not know which particle is to be measured and which one to be used as the embedding source. Hence, it gathers no further useful insights;
  - *Intercept and Resend Particles*: An attacker might try to measure all v photons and then re-encode the information, performing a *man-in-the-middle* attack. However, the security constraints are analogous to those of the *BB84* protocol. Without prior knowledge of the encoding basis, the attacker has a 25% chance of guessing the correct measurement basis for each particle. For a stream of $v-1$ particles, where the entangled particle must not be measured, the probability of correctly guessing all measurement bases is $\frac{1}{4}^{(v-1)}$, making the attack highly improbable.
- *Spoofing*: to compromise the protocol by spoofing a participant's identity, an attacker would need to obtain a *private key* or break the *CRYSTALS-Kyber* primitives. Currently, no known methods exist to easily solve the lattice problems underlying this encryption scheme, making such an attack highly impractical.
- *Collusion*: Given n *players*, 1 *dealer*, t participating hosts, f fraudulent players (with $f < t$) providing $M'_j, j \in \{0, \dots, f-1\}$ fake measurements they are able to fix the result iff:

$$\begin{gathered} \sum_{i=0}^{t} M_i = S \\ \Rightarrow \sum_{i=0}^{t-1-f} M_i + \sum_{j=0}^{f-1} M'_j = S' \\ H(S) = H(S') \end{gathered} \tag{19}$$

  Hence, a sub-group of rogue *players* can threaten the protocol if, and only if, they are able to create an hash-collision with their fake measurements. In this case, only the malicious *players* would know the real protocol's output.
- *Trojan Horse*: Given that the proposed protocol introduces a BB84-like photon exchange to conceal the entangled particle and enhance rogue node detection, it is essential to consider the potential for a *Trojan-Horse* attack scenario. This issue becomes even more significant when there are one or more relay nodes between the *dealer* and the *player*, as this impacts the protocol's security.
  Ideally, all light entering a receiving system would seamlessly transfer to the output via interfaces and components. However, in real-world scenarios, some light may inevitably be reflected or scattered back while passing through an interface. Fresnel reflections occur due to variations in the refractive index during propagation, while Rayleigh or Brillouin scattering results from density fluctuations in the optical fiber material. The wavelength and intensity of the incoming light affect the amount of scattering and reflection. An eavesdropper might introduce a light pulse via the quantum channel into a Quantum Key Distribution (QKD) subsystem, encountering multiple sites of reflection and scattering. This could allow the attacker to gather information on how the sender polarized the last emitted particle.
  While it might seem overzealous to discuss additional security measures given the QKD-like method used in the GHZ states' sharing phase of the protocol, it is important to note that even though the states' polarization does not carry any information, detecting attempts to

mine the exchange is crucial. Since relays are part of the quantum-network definition, enhancing security also involves adding layers to detect unfair behavior.

To increase security against *Trojan-Horse* attacks, Fig.14, one approach is to minimize the emitter's *opening-frame* time to reduce the eavesdropper's time-span. To block and detect an attack, a combination of an *isolator* and a *watchdog* can be used. This setup ensures that incoming pulses are either blocked or detected, alerting the *players* and identifying the rogue node. Additionally, an optical filter (such as Bragg gratings (FBGs) or Fabry-Perot cavities) can be added after the monitoring detector. Physical constraints, such as using angle polished connectors (FC/APC) instead of flat connectors (FC/PC), can also reduce light back-scattering, further enhancing security against potential *Trojan-Horse* attacks.

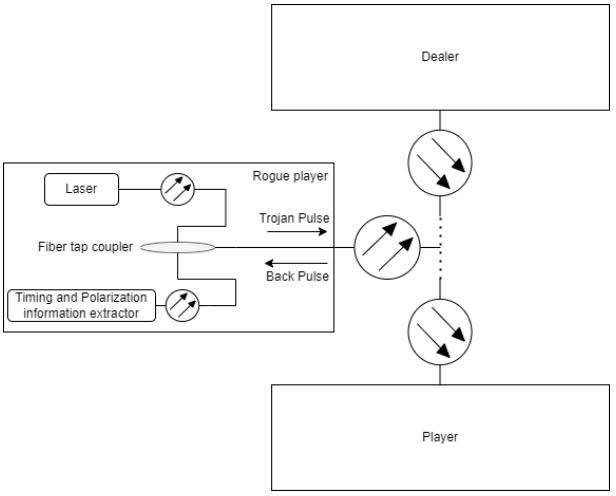

FIGURE 14: Trojan Horse attack implementation [40].

VI. CONCLUSIONS

This work has presented a novel entanglement-based QSS protocol that enhances the flexibility and resilience of quantum network topologies overcoming some limitations of existing models. By ensuring that players remain unaware of their collaborators and the network structure, the protocol mitigates risks of collusion, making it highly suitable for deployment in secure environments.

A central innovation lies in the use of the Quantum-Dijkstra algorithm for selecting the subset of t players, introducing an efficient and adaptable method for participant selection. This algorithm accounts for practical considerations such as fiber propagation properties and proximity to the dealer, ensuring optimal performance in dynamic quantum networks. Moreover, the integration of CRYSTALS-Kyber, a post-quantum cryptographic scheme, for player authentication further strengthens the protocol's security framework by ensuring that only authorized participants are involved, reducing risks of identity spoofing and enhancing trust.

The protocol emphasizes fairness within a (t, n)-scheme, rather than the traditional (n, n)-scheme, providing a robust mechanism to verify the correctness of results and detect tampering attempts. This approach reinforces the integrity of the protocol against potential cheating, while the comprehensive application of the extended CIA Triad ensures a robust framework for information security. Together, these attributes address a wide range of security threats, demonstrating a holistic approach to safeguarding quantum communications.

The results of numerical and practical simulations confirm the real-world viability of the proposed protocol, while demonstrating the protocol's ability to embed multi-level arithmetic in qubit-based circuits and verifying its correctness across various threshold settings and multi-hop routes. On actual hardware, despite current constraints imposed by noise models and two-level systems, observed success rates validate the protocol's cryptographic assurances. With further advancements in error mitigation strategies and mid-circuit measurement technologies, this protocol holds strong potential for deployment in realistic network configurations. The proposed entanglement-based QSS protocol demonstrates a rigorous and balanced approach to advancing the security and applicability of quantum communication networks. By addressing both immediate and future challenges, it paves the way for more secure, efficient, and adaptable quantum information sharing, fostering confidence in the practical realization of quantum technologies.

Although representing a significant step forward, this work also underscores key avenues for future research. In the realm of post-quantum cryptography, the reliance of schemes like CRYSTALS-Kyber on the hardness of the Learning with Errors (LWE) problem highlights the need for continued study. Though LWE currently underpins robust security against quantum attacks, the possibility of breakthroughs in quantum algorithms—akin to Shor's Algorithm for factoring—cannot be discounted. Researchers must remain vigilant, exploring alternative cryptographic foundations and preparing for potential advancements in quantum computation.

Similarly, quantum cryptography offers a fertile ground for innovation. While this study adapted classical algorithms, such as Dijkstra, into the quantum realm, future efforts could explore the development of inherently quantum concepts, unbound by classical constraints. Protocols like BB84 and E91 exemplify how native quantum approaches can achieve unparalleled security. As confidence in quantum cryptographic primitives grows, governments and industries will need to accelerate their adoption of these technologies to safeguard data in the post-quantum era.

ACKNOWLEDGMENT

This work is supported by the project ISP5G+ (CUP D33C22001300002), which is part of the SERICS program (PE00000014) under the NRRP MUR program funded by the EU-NGEU.

REFERENCES

[1] G. R. Blakley. Safeguarding cryptographic keys. 1979 International Workshop on Managing Requirements Knowledge (MARK), pages 313–318, 1979.

[2] A. Shamir. How to share a secret. Communications of the ACM, 22(11):612–613, 1979.

[3] P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Journal on Computing, 26(5):1484–1509, Oct. 1997.

[4] C. Lu, F. Miao, J. Hou, W. Huang, and Y. Xiong. A verifiable framework of entanglement-free quantum secret sharing with information-theoretical security. Quantum Information Processing, 19(1), 2019.

[5] B. C. Hiesmayr. Free versus bound entanglement, a np-hard problem tackled by machine learning. Scientific Reports, 11(1), 2021.

[6] R. van Houte, J. Mulderij, T. Attema, I. Chiscop, and F. Phillipson. Mathematical formulation of quantum circuit design problems in networks of quantum computers. Quantum Information Processing, 19(5), 2020.

[7] F. Liu, S.-J. Qin, and Q.-Y. Wen. A quantum secret-sharing protocol with fairness. Physica Scripta, 89(7):075104, 2014.

[8] M. Hillery, V. Bužek, and A. Berthiaume. Quantum secret sharing. Phys. Rev. A, 59(3):1829–1834, 1999.

[9] Daniel M. Greenberger, Michael A. Horne, and Anton Zeilinger. Going beyond bell's theorem, 2007.

[10] H.-Y. Jia, Q.-Y. Wen, F. Gao, S.-J. Qin, and F.-Z. Guo. Dynamic quantum secret sharing. Physics Letters A, 376(10-11):1035–1041, 2012.

[11] C.-H. Liao, C.-W. Yang, and T. Hwang. Dynamic quantum secret sharing protocol based on GHZ state. Quantum Information Processing, 13(8):1907–1916, 2014.

[12] J. L. Hsu, S. K. Chong, T. Hwang, and C. W. Tsai. Dynamic quantum secret sharing. Quantum Information Processing, 12(1):331–344, 2013.

[13] S. Mishra, C. Shukla, A. Pathak, R. Srikanth, and A. Venugopalan. An integrated hierarchical dynamic quantum secret sharing protocol. International Journal of Theoretical Physics, 54(9):3143–3154, 2015.

[14] H. Qin and Y. Dai. Dynamic quantum secret sharing by using d-dimensional ghz state. Quantum Information Processing, 16(3):64, 2017.

[15] Z. You, Y. Wang, Z. Dou, J. Li, X. Chen, and L. Li. Dynamic quantum secret sharing between multiparty and multiparty based on single photons. Physica A: Statistical Mechanics and its Applications, 624:128893, 2023.

[16] P. Priyanka, V. Siwach, and P. Bijaranian. Quantum secret sharing with (m, n) threshold: Qft and identity authentication, 2024.

[17] Nanrun Zhou, Zhenyong Chen, Yanyan Liu, and Lihua Gong. Multi-party semi-quantum private comparison protocol of size relation with d-level ghz states. Advanced Quantum Technologies, 2024.

[18] Lihua Gong, Min Li, Hong Cao, and Bo Wang. Novel semi-quantum private comparison protocol with bell states. Laser Physics Letters, 21(5):055209, 2024.

[19] W. Hu, R.-G. Zhou, X. Li, P. Fan, and C. Tan. A novel dynamic quantum secret sharing in high-dimensional quantum system. Quantum Information Processing, 20(5):159, 2021.

[20] Y. Song, Z. Li, and Y. Li. A dynamic multiparty quantum direct secret sharing based on generalized ghz states. Quantum Information Processing, 17(9):244, 2018.

[21] C.-W. Yang and C.-W. Tsai. Improved dynamic multiparty quantum direct secret sharing protocol based on generalized ghz states to prevent collusion attack. Modern Physics Letters A, 35(8):2050040, 2020.

[22] Y. Tian, J. Wang, G. Bian, J. Chang, and J. Li. Dynamic multi-party to multi-party quantum secret sharing based on bell states. Advanced Quantum Technologies, 7(7):2400116, 2024.

[23] Y. Kang, Y. Guo, H. Zhong, G. Chen, and X. Jing. Continuous variable quantum secret sharing with fairness. Appl. Sci., 10(1):189, 2020.

[24] X. Li, K. Zhang, L. Zhang, and X. Zhao. A new quantum multiparty simultaneous identity authentication protocol with the classical third-party. Entropy, 24(4):483, 2022.

[25] S. Schauer, M. Huber, and B. C. Hiesmayr. Experimentally feasible security check for n-qubit quantum secret sharing. Phys. Rev. A, 82(6), 2010.

[26] G. Gao, C.-C. Wei, and D. Wang. Cryptanalysis and improvement of dynamic quantum secret sharing protocol based on two-particle transform of bell states. Quantum Information Processing, 18(6):186, 2019.

[27] Ryan Golden and Ilwoo Cho. On symmetric polynomials, 2015.

[28] K. Thas. The geometry of generalized pauli operators of n-qudit hilbert space, and an application to mubs. Europhysics Letters, 86(6):60005, jul 2009.

[29] C. H. Bennett and G. Brassard. Quantum cryptography: Public key distribution and coin tossing. Theor. Comput. Sci., 560:7–11, 2014.

[30] F. Li, T. Chen, and S. Zhu. A (t,n) threshold quantum secret sharing scheme with fairness. Int. J. Theor. Phys., 62(6):119, 2023.

[31] W. Liu, Q. Wu, J. Shen, J. Zhao, M. Zidan, and L. Tong. An optimized quantum minimum searching algorithm with sure-success probability and its experiment simulation with cirq. Journal of Ambient Intelligence and Humanized Computing, 12(11):10425–10434, Jan. 2021.

[32] G. L. Long. Grover algorithm with zero theoretical failure rate. Phys. Rev. A, 64(2), 2001.

[33] E. W. Dijkstra. A note on two problems in connexion with graphs. Numerische Mathematik, 1(1):269–271, Dec 1959.

[34] Z. Ji, P. Fan, and H. Zhang. Entanglement swapping theory and beyond, 2022.

[35] X. Gitiaux, I. Morris, M. Emelianenko, and M. Tian. Swap test for an arbitrary number of quantum states, 2021.

[36] M. Shtaif, C. Antonelli, A. Mecozzi, and X. Chen. Challenges in estimating the information capacity of the fiber-optic channel. Proceedings of the IEEE, 110(11):1655–1678, 2022.

[37] J. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. M. Schanck, P. Schwabe, G. Seiler, and D. Stehle. Crystals - kyber: A cca-secure module-lattice-based kem. In 2018 IEEE European Symposium on Security and Privacy (EuroS&P), pages 353–367, 2018.

[38] Ali Javadi-Abhari, Matthew Treinish, Kevin Krsulich, Christopher J. Wood, Jake Lishman, Julien Gacon, Simon Martiel, Paul D. Nation, Lev S. Bishop, Andrew W. Cross, Blake R. Johnson, and Jay M. Gambetta. Quantum computing with Qiskit, 2024.

[39] Alessio Di Santo. Security and fairness in multi-party quantum secret sharing protocol. https://github.com/alessiobb3b/Security-and-Fairness-in-Multi-Party-Quantum-Secret-Sharing-Protocol, 2025.

[40] Y. Pan, L. Zhang, and D. Huang. Practical security bounds against trojan horse attacks in continuous-variable quantum key distribution. Appl. Sci., 10(21):7788, 2020.

(None of the authors have a conflict of interest to disclose)

ALESSIO DI SANTO (Graduate Student Member, IEEE) received a Bachelor's degree in Information Engineering in 2020 from the Università degli Studi dell'Aquila, with a thesis focused on fairness and cryptography. In 2022, He completed a Master's degree at the same institution, presenting a thesis on forensic acquisition techniques for Windows IT/OT assets. Currently, he is pursuing a Ph.D. at the Università degli Studi dell'Aquila under the supervision of Professor Dajana Cassioli, with co-tutor Walter Tiberti. Since 2020, he has been employed in the cybersecurity sector, working as a Cyber Threat Intelligence Analyst, Incident Responder, and Malware Analyst.

WALTER TIBERTI (Member, IEEE) received the B.Sc. and M.Sc. degrees in computer engineering and the Ph.D. degree in information and communications technology from the University of L'Aquila, working on embedded systems security, in particular, wireless sensor network security. He is currently an Assistant Professor with the Department of Information Engineering, Computer Science and Mathematics, University of L'Aquila. He was a Visiting Researcher with the Research Centre in Real-Time and Embedded Computing Systems (CISTER), Porto, Portugal. He is part of the technical committee member of IEEE conferences. He is involved in Italian initiatives to promote nationwide cyber-security education and training. His research work is focused on software security, network security, and cryptography.

DAJANA CASSIOLI (Senior Member, IEEE) is currently an Associate Professor in telecommunications engineering with the University of L'Aquila, Italy. Her research interests include wireless communications, 5G/B5G networks, and cybersecurity. She is also the Chair of the IEEE ComSoc RCC SiG on Propagation Channels for 5G&B and the Diversity, Equity, and Inclusion Activity Coordinator of the IEEE Italy Section. She is the Past Chair of the IEEE WIE AG Italy Section (2016–2022) and IEEE VT06/COM19 Italy Chapter (2011–2017). Since 2015, she has been the Coordinator of the University of L'Aquila Node, CINI National Laboratory of Cybersecurity, where she led the CyberEquality WG (2020–2021). She has been awarded the ERC StG VISION (Video-Oriented UWB-Based Intelligent Ubiquitous Sensing), in 2010, and the ERC PoC Grant MobIle health-Care system for monitoring toxicity and symptoms in cAncer patients Receiving diseasE-oriented therapy (iCARE), in 2016. She was the CEO (2014–2018) and in 2019, of the spin-off of the University of L'Aquila, Smartly: Natives of Smart Living Srl. She served as the IEEE EUROCON 2023 WIE Chair, the IEEE ICC 2023 CISS Co-Chair, the PIMRC2018 Industry Co-Chair, the RTSI WIE Chair, in 2018, 2019, and 2020, MELECON2020 and MetroInd 4.0, and a TPC Member of several international conferences (ICC, PIMRC, VTC, and GLOBECOM). She is also an Associate Editor of Electronic Letters (IET) and IEEE COMMUNICATIONS LETTERS and an Executive Editor of Internet Technology Letters (Wiley) and Transactions on Emerging Telecommunication Technologies. In 2000, she was the Summer Manager of the Wireless Systems Research Department, AT&T Laboratories-Research, NJ, USA. She participated in defining the standard channel model for the IEEE 802.15.4 standard, in 2005. In 2022, she was a Visiting Short-Term Scholar with the University of Southern California, LA, USA.