applied sciences

IMPACT FACTOR 2.7

CITESCORE 4.5

Article

CPSR-HQKDN: A Hybrid Trusted Relay Quantum Key Distribution Network Routing Scheme Based on Classification of Packet Security Requirements

Lin Bi, Weijie Wu, Xiaotong Yuan, Minghui Miao, Xiaoqiang Di and Zhengang Jiang

MDPI

https://doi.org/10.3390/app132212284

applied sciences

MDPI

Article

CPSR-HQKDN: A Hybrid Trusted Relay Quantum Key Distribution Network Routing Scheme Based on Classification of Packet Security Requirements

Lin Bi [1,2], Weijie Wu [1,2,*], Xiaotong Yuan [1,2], Minghui Miao [1,2], Xiaoqiang Di [1,2] and Zhengang Jiang [1,2]

[1] School of Computer Science and Technology, Changchun University of Science and Technology, Changchun 130012, China
[2] Key Laboratory of Network and Information Security in Jilin Province, Changchun 130012, China
* Correspondence: m15906196992@163.com

Abstract: To ensure the security of information exchange in software-defined optical networks, quantum key distribution (QKD) based on quantum mechanics is introduced. However, the slow and valuable process of generating quantum key resources contradicts the high-speed data transmission requirements of optical networks. To address this issue, this paper proposes the CPSR-HQKDN scheme, which takes into account factors such as security requests, key demand, key residual, and key update rates for trusted and untrusted links. This approach improves resource utilization and service efficiency by optimizing the processing order of key requests. Moreover, the routing strategy dynamically adjusts based on the network resource environment, thereby increasing the success rate of key requests. Through simulation experiments comparing the performance of the CPSR-HQKDN routing scheme with existing schemes, it is observed that in high-concurrent scenarios, the CPSR-HQKDN routing scheme can improve the success rate of key requests by at least 5%.

Keywords: packet security requirements; routing scheme; quantum key distribution (QKD)

Citation: Bi, L.; Wu, W.; Yuan, X.; Miao, M.; Di, X.; Jiang, Z. CPSR-HQKDN: A Hybrid Trusted Relay Quantum Key Distribution Network Routing Scheme Based on Classification of Packet Security Requirements. *Appl. Sci.* **2023**, *13*, 12284. https://doi.org/10.3390/app132212284

Academic Editor: Marco Genovese

Received: 3 October 2023
Revised: 30 October 2023
Accepted: 7 November 2023
Published: 13 November 2023

1. Introduction

With the continuous development of network and communication technologies, there is inevitably a large amount of private and sensitive information being transmitted in the existing network communication environment. However, the emergence and constant breakthroughs of quantum computing technology and quantum computers pose a threat to the security of data protection in traditional cryptography [1,2]. Traditional encryption algorithms mainly rely on the difficulty of mathematical problems, such as factorization and discrete logarithms. The attack of quantum computing on traditional encryption algorithms is primarily based on the quantum parallelism and quantum search algorithms possessed by quantum computers. Traditional encryption algorithms like RSA encryption are secured based on the difficulty of factorizing large numbers, where one important step is to choose two large prime numbers as part of the private key [3]. Classical computers currently lack the efficient means to factorize large numbers. However, in quantum computing, Shor's algorithm is a quantum algorithm proposed by Peter Shor [4], which can efficiently perform factorization on quantum computers. By utilizing Shor's algorithm, quantum computers can quickly factorize the large numbers used in RSA encryption and thus break the RSA encryption algorithm [5]. Similarly, Elliptic Curve Cryptography (ECC) is a commonly used symmetric encryption algorithm in modern cryptography, which exploits the difficulty of operations on points on elliptic curves [6]. Traditional computers require a long time to break ECC encryption; however, Grover's algorithm is a quantum search algorithm that can accelerate the search for solutions on quantum computers. By applying Grover's algorithm, quantum computers can effectively reduce the search space and find the private key of the ECC encryption algorithm in a shorter time, thereby breaking the encryption [7].

To counter the threat of quantum computing to traditional cryptography, two new encryption methods have emerged: Post-Quantum Cryptography (PQC) and Quantum Key Distribution (QKD) [8–10]. PQC is different from traditional encryption algorithms based on mathematical problems as it mainly relies on other difficult problems such as lattice problems, polynomial ring problems, and coding problems. The goal of PQC is to provide security comparable to that of traditional encryption algorithms and to withstand attacks by quantum computers [8]. QKD is a method of using quantum mechanics principles to achieve encrypted key transmission. As one of the most successful applications of quantum cryptography, QKD promises information-theoretic security based on quantum physics laws for distributing symmetric keys between a pair of legitimate parties. Then, symmetric key cryptographic systems can use these keys to encrypt confidential messages transmitted over a public channel. One example of symmetric key cryptographic systems is the so-called one-time pad (OTP), which Shannon has proved to help in secure message encryption in information theory. However, its disadvantage is that the key must be at least as long as the message. Of course, shorter keys can also be used through some methods; other symmetric key cryptographic systems, such as the Advanced Encryption Standard (AES), are also considered quantum-safe. A key challenge faced by symmetric key cryptographic systems is secure key sharing, which QKD can bypass. In recent years, point-to-point QKD technology has made significant progress in protocols, devices, systems, and so on to improve QKD performance, which can be quantified in terms of the key rate, distance, and security. Therefore, QKD systems have been commercially available in the market. With the arrival of the 5G communication era and the continuous development of 6G communication technology, quantum networks built with related quantum communication technologies such as QKD are likely to become an important part of future networks [11–13].

Among them, the BB84 protocol [14] and the MDI-QKD protocol [15] are representative protocols in quantum key distribution. The BB84 protocol utilizes the principles of quantum mechanics to achieve secure key distribution by sending and detecting quantum bits. Based on two mutually orthogonal quantum states, the BB84 protocol detects the presence of potential eavesdroppers (often referred to as Eve) by exploiting the properties of quantum states, and verifies the correctness of the bits through a public classical channel. Ultimately, Alice and Bob can negotiate a shared secret key by eliminating errors and eavesdropping. The MDI-QKD (Measurement Device Independent Quantum Key Distribution) protocol is an advanced quantum key distribution protocol that offers higher security and robustness. Unlike traditional QKD protocols, the MDI-QKD protocol does not require Alice and Bob to trust each other's measurement devices. The MDI-QKD protocol eliminates the need for device trust by introducing auxiliary photons shared by both signaling parties for quantum state measurements. In the MDI-QKD protocol, Alice and Bob communicate using their own quantum signals carryings quantum bits (typically polarization states) and perform joint measurements using a shared auxiliary photon. Through public interactions over a classical channel, Alice and Bob can extract a secure key.

Currently, quantum key distribution (QKD) technology still has some limitations in certain aspects. Firstly, generating quantum bits in quantum key distribution is challenging, as it involves manipulating and measuring quantum states. Especially for long-distance quantum links, maintaining the integrity and stability of quantum bits becomes more difficult and requires complex technical means and devices. Secondly, the implementation of QKD technology heavily relies on physical devices and laboratory environments, which are costly and difficult to deploy on a large scale. Therefore, quantum keys produced through QKD are precious and rare compared to traditional keys. Lastly, it is worth studying how to design network architectures and algorithms to improve the utilization efficiency of quantum keys [16,17].

2. Problem Statement

Several researchers have proposed various approaches and solutions for quantum network construction. In terms of the physical layer and communication, Shuang Wang

and Zhen-Qiang Yin et al. proposed an experimental QKD system that can tolerate channel losses exceeding 140 dB and achieve a secure distance of 833.8 km, setting a new record for fiber-based QKD [18]. Wei Zhang and Tim van Leent et al. presented an experimental system that realized DI-QKD based on entangled states of radium atoms prepared and analyzed at a distance of 400 m within a building, demonstrating the capability of generating keys in the system [19].

Researchers in the academic community have also actively participated in related studies in the field of networking. M. POMPILI and S.L.N. HERMANS proposed a three-node remote quantum network based on the photonic coupling of solid-state spin qubits [20]. Zhantong Qi and Yuanhua Li et al. explored a QSDC network based on time–energy entanglement and frequency conversion, which included a fully connected QSDC network with 15 users. Experimental results demonstrated the feasibility of this QSDC network, laying a foundation for satellite-based remote QSDC implementation [21]. SIDDARTH KODURU JOSHI et al. presented a fully connected quantum communication metropolitan area network without active switching or trusted nodes. This network can easily be scaled to accommodate more users while minimizing the required infrastructure and hardware, thus reducing network construction costs [22]. Yuan Cao and Yongli Zhao et al. investigated the deployment of QKD with hybrid trusted/untrusted relay nodes based on an optical backbone network. They designed an integer linear programming model and a heuristic algorithm to optimize deployment costs while maintaining a higher level of security compared to traditional point-to-point QKD protocols with only trusted relay nodes [23]. Furthermore, Yuan Cao authored a review article to introduce the fundamental knowledge of quantum key distribution (QKD) and review the development of QKD networks as well as their implementation in practice. Subsequently, the general architecture, components, interfaces, and protocols of QKD networks were described, and relevant physical layer and network layer solutions were summarized, providing guidance for the design of QKD networks [24].

In network research, routing is a critical problem. Weike Ma and Bowen Chen investigated the performance of balancing the allocation of quantum key resources in optical data center networks using Quantum Key Distribution (QKD). In order to effectively utilize quantum key resources, they proposed three efficient load balancing routing, wavelength, and time-slot allocation (LB-RWTA) methods, which have high efficacy in quantum key resource allocation and network performance management [25]. Xiaosong Yu and Yuhang Liu et al. studied the key routing in different typical network topologies under partially trusted relay scenarios and proposed a secret key distribution (SKP)-based collaborative routing algorithm for finding the optimal key relay routing path. The experimental results show that compared with traditional schemes based on trusted relays, the SKP-CR algorithm has significant advantages in terms of a key distribution success rate in a mesh topology [26].

This article presents a novel approach that differs from traditional network schemes utilizing a single quantum key distribution (QKD) protocol. Instead, it proposes a hybrid QKD network model consisting of two links, namely the BB84 protocol and the MDI-QKD protocol. Furthermore, this approach establishes a unique linear programming mathematical model based on the key bit generation characteristics of BB84 and MDI-QKD. A dynamic routing algorithm is designed based on the remaining key quantities at each node, and a new queuing theory for encrypted data packets is integrated to alleviate the overall congestion in the quantum network. This approach aims to reduce the packet loss rate and improve quantum key utilization.

3. Network Model

3.1. Quantum Key Distribution Luminescence Network Architecture Based on Hybrid Trusted Relay

In quantum key distribution networks, the use of relay nodes is a promising solution for achieving remote key distribution. This includes quantum relays, trusted relays, and

untrusted relays. The principle of quantum relays is to use quantum entanglement to generate entangled states between two distant nodes, thereby establishing secure long-distance communication [27]. However, quantum relay technology is still immature and costly at present. Therefore, the current approach is to use trusted relay nodes for hop-by-hop communication to generate quantum keys between adjacent nodes, such that large-scale QKDNs are mainly composed of end-users and trusted nodes. However, as the complexity of applications and the number of nodes increase, the security and reliability of the network also face challenges [28]. Finally, researchers introduce untrusted relay nodes as a solution. By using the MDI-QKD protocol, which has device independence and high transmission efficiency, the overall performance of QKDNs can be significantly improved. Thus, we propose a hybrid trust relay-based QKDN scheme (HQKDN), and combine it with the Software Define Network (SDN) framework to achieve the separation of network control and data forwarding operations for flexible network resource management [29].

As shown in Figure 1, the specific architecture of the hybrid Quantum Key Distribution (QKD) network based on Software-Defined Networking (SDN) is presented, and the functional modules at each layer are described in detail below.

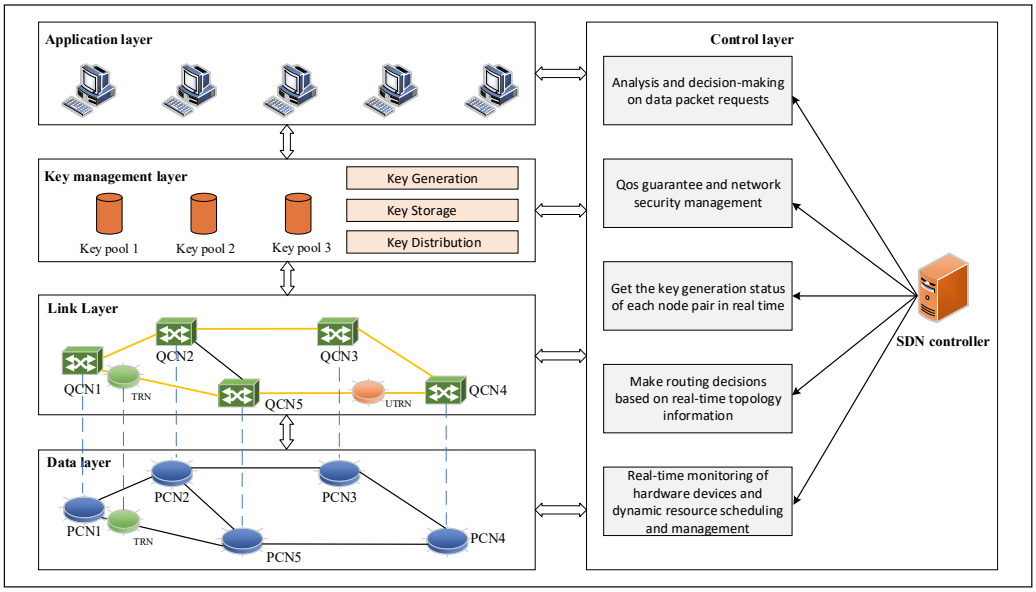

Figure 1. The architecture diagram of a five-layer hybrid trusted relay QKDN based on SDN.

Data Layer: The data layer is primarily responsible for data transmission and forwarding in optical networks. It achieves high-speed data transmission in optical fibers by scheduling and controlling the transmission paths of optical signals. It also performs the packetization and forwarding of data according to network traffic requirements.

QKD Link Layer: The QKD layer is the core layer responsible for implementing quantum key distribution technology. It is responsible for generating, transmitting, and receiving quantum keys while ensuring their security and integrity. In this layer, the BB84 protocol and MDI-QKD protocol are applied for the generation and distribution of quantum keys.

Key Management Layer: The key management layer is responsible for managing and distributing the generated quantum keys. It encompasses functions such as key storage, updating, verification, and distribution. In SDN-based optical networks, the key management layer needs to closely collaborate with the network control layer to ensure the secure management and effective utilization of the keys.

Control Layer: The control layer serves as the core of SDN-based optical networks, responsible for network control and management. It programs and configures the entire network through a centralized controller, enabling the comprehensive management of the data layer, QKD layer, and key management layer. In the context of QKD technol-

ogy, the control layer coordinates and controls the process of quantum key distribution while engaging in information exchange with other layers to ensure network security and performance.

Application Layer: The application layer serves as the primary service entity in a QKD network, acting as a bridge between the QKD network and real users. Within this layer, application programs initiate key requests, which are then submitted to the controller. Upon receiving a key request, the application program enters a blocking and waiting phase. Only when the application program receives a response message from the controller can it proceed to the subsequent phase of key generation. The application layer is responsible for defining and implementing high-level protocols and applications required for secure communication. It also encompasses the user–network interaction interface, as well as specific business requirements and application scenarios for secure communication. These functionalities are achieved through high-level application interfaces and can be extended with various features based on different application needs.

3.2. QKD Protocol, Model and Related Definitions

Within the entire network architecture of the CPSR-HQKDN scheme, Quantum Key Distribution (QKD) protocols play a crucial role in data packet encryption [30]. This scheme employs a collaborative deployment of the BB84 protocol and the MDI-QKD protocol to allocate distinct paths for data packets based on varying security requirements. Illustrated in Figures 2 and 3, the basic principles of point-to-point communication using the BB84 protocol and MDI-QKD protocol in the Quantum Key Distribution (QKD) links are presented.

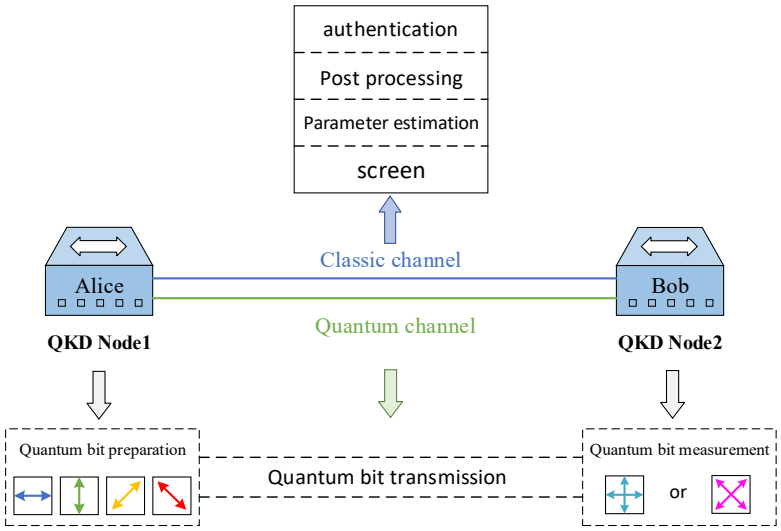

Figure 2. Basic principle diagram of using the BB84 protocol in the QKD link.

The BB84 Protocol: Firstly, both parties need to prepare a key. The sender (Alice) and the receiver (Bob) each prepare a key, which can be generated randomly as a bit sequence (e.g., 01 sequence). Then, Alice sends a sequence of quantum bits through a quantum channel to Bob, where each bit represents a randomly chosen quantum state, such as horizontally/vertically or diagonal/anti-diagonal polarized photons. Upon receiving the quantum bits, Bob randomly chooses a set of measurement bases. Each measurement basis corresponds to a different polarization direction of the quantum bit. Bob measures the received quantum bits using the selected measurement bases and records the outcomes. It should be noted that due to the nature of quantum states, the measurement outcomes are affected by uncertainty. Bob then publicly announces to Alice the information about the selected measurement bases, but does not disclose the specific measurement outcomes. At this point, Alice and Bob confirm through classical communication that they have indeed used the same measurement bases. This step is crucial to ensure the subsequent comparison

of the bit values' meaning. Alice and Bob compare their measured outcomes, and based on the previously confirmed measurement basis information, they determine which outcomes are trustworthy. These trustworthy outcomes are used to generate the key. Next, Alice and Bob calculate the bit-error rate based on the publicly compared partial bit values. If the error rate exceeds a specific threshold, there may be a risk of eavesdropping or communication errors. This will determine whether the current key is retained. Finally, Alice and Bob extract the final key through an error correction protocol. This protocol can be achieved by using error correction codes to ensure the security of the key, even at a limited error rate. After obtaining the key, Alice and Bob can store it in the key pool [31].

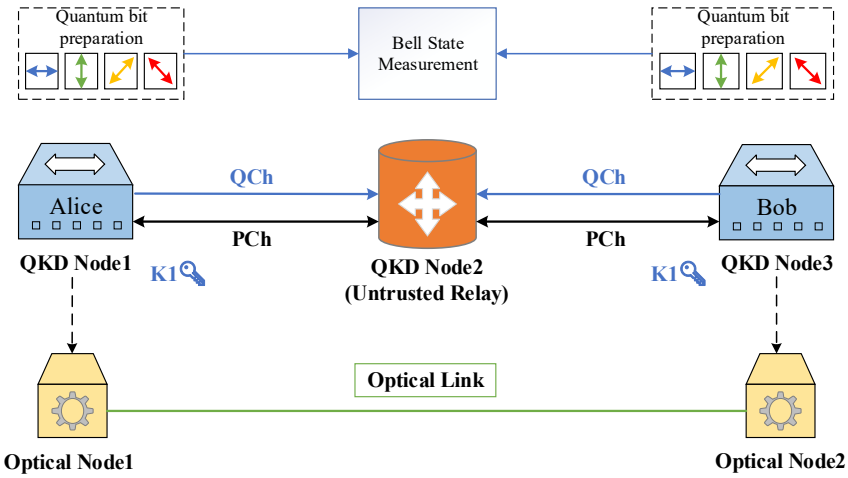

Figure 3. Basic principle diagram of using the MDI-QKD protocol in the QKD link.

MDI-QKD Protocol: As shown in Figure 3,Alice and Bob use phase randomization to prepare four BB84 states with weak coherent light pulses, including horizontal polarization state $|H\rangle$, vertical polarization state $|V\rangle$, diagonal polarization state $|+\rangle = \sqrt{2}(|H\rangle+|V\rangle)$, and $|-\rangle = \sqrt{2}\,(|H\rangle-|V\rangle)$. They send these states through a quantum channel to a third party, Charlie, and announce the encoding basis they used. Charlie performs Bell state measurements after receiving two qubits and publicly announces the successful measurement results. Meanwhile, Alice and Bob select to flip or not flip their bits on hand for the parts where they use the same basis as Charlie's to obtain positively correlated data. For example, if Charlie measures $|HV\rangle$, Alice and Bob choose to flip their vertical polarization state $|V\rangle$. To obtain the gain and error rate of the single-photon portion, Alice and Bob use the decoy-state method. They each choose a random basis and send it to each other through the quantum channel. After receiving the basis, the other party measures its polarization state and informs the result. Based on these results, Alice and Bob can calculate the gain and error rate of the single-photon portion. After obtaining the gain and error rate, classical error correction techniques can be applied to correct the erroneous bits, and privacy amplification techniques can be used to improve the security of the key. Finally, Alice and Bob obtain a set of secure keys that can be used for secure communication. Through steps such as phase randomization, Bell state measurement, decoy-state method, classical error correction, and privacy amplification, the security and reliability of the key can be guaranteed [32].

The BB84 protocol is based on the principle of the unclonability of quantum states, which ensures that the key information cannot be obtained by a middleman due to the characteristic that quantum states cannot be copied or stolen [33]. The security of the MDI-QKD protocol is based on the impossibility of Bell state measurements, which means that a middleman cannot simultaneously interfere with and eavesdrop on the two quantum states sent by Alice and Bob. In contrast, in the BB84 protocol, a middleman can more easily steal communication between Alice and Bob and tamper with data. By utilizing the quantum entanglement properties of Bell states, the MDI-QKD protocol effectively avoids

middleman attacks and eavesdropping attacks, thus providing higher attack resistance and security. Based on these principles, the MDI-QKD protocol can achieve higher security and confidentiality in quantum communication. On the contrary, due to greater instability, higher deployment costs, and lower key generation rates under the same link deployment length conditions, the keys generated by the MDI-QKD protocol are more valuable than those generated by the BB84 protocol [34].

The CPSR-HQKDN scheme is suitable for scenarios with varying security requirements for the data packets to be encrypted and significant differences in key demands. It establishes priority criteria for different requests. Therefore, this paper employs a QKD network model to define routing strategies for packet encryption. Additionally, the total key demand and key update rate of the QKD network are evaluated. The following terms are defined in the model:

1. QKD Link: A QKD link refers to the physical components involved in quantum key distribution, including optical transmitters, optical receivers, and the transmission medium. It is a virtual link abstracted between two quantum nodes, where the transmission and measurement of quantum bits based on QKD technology take place on the channel.
2. Trusted Relay/Untrusted Relay: A trusted relay is an entity trusted by Alice and Bob, capable of establishing a secure communication link between them. Its functions mainly include channel enhancement, relay authentication, and key generation and distribution. Therefore, a trusted relay involves operations such as key storage and key utilization with the key pool. In MDI-QKD, an untrusted relay is considered an entity that cannot be trusted. Its functions mainly include the forwarding of quantum states, the measurement of Bell states, and noise filtering.
3. Quantum Key Pool: The quantum key pool is a virtual concept that represents a series of quantum keys generated between a pair of quantum nodes in time slots. It also involves the issue of how to allocate keys to different data packets within the same time slot.
4. Data Packet Security Requirement Levels: During the process of data transmission, not all data packets have the same security requirements. Some data packets may only require traditional encryption methods or even no encryption at all. Some data packets may be encrypted using quantum keys generated by the BB84 protocol. For the data packets with the highest security requirements, they will be encrypted using more valuable and secure quantum keys generated by the MDI-QKD protocol.
5. Path Constraints: Different path selection constraints exist for data packets with different security requirements. First, for data packets that only require traditional encryption methods or no encryption at all, they do not need to pass through the QKD layer and can be directly processed at the traditional data layer. Therefore, they can choose the shortest path. Second, data packets encrypted using quantum keys generated by the BB84 protocol can only pass through backbone nodes and trusted relay nodes that utilize the BB84 protocol. Finally, data packets encrypted using quantum keys generated by the MDI-QKD protocol can only pass through backbone nodes, pairs of trusted relays, and untrusted relays located between the pairs of trusted relays that utilize the MDI-QKD protocol.
6. Time Slot: In the field of communications, a time slot is a unit of time used for the allocation and management of resources in multiplexing techniques. Time slots divide time into non-overlapping intervals, with each time slot designated for the transmission of specific data or signals. One of the advantages of using time-division multiplexing is the efficient utilization of transmission resources and shared bandwidth. By allocating time slots appropriately, different users or channels can transmit data simultaneously over the same physical link, thereby improving transmission efficiency. In the CPSR-HQKDN scheme, time slots primarily serve the purpose of calculating the remaining quantity of keys and various key rates, among others, in preparation for addressing the routing issues of data packets.

7. Key Update Rate: In QKD technology, the key update rate refers to the number of newly generated key bits successfully generated within a unit of time. In this scheme, the key update rate is calculated based on the total number of key bits successfully generated and the total time taken.
8. Packet Encryption Request: Within the SDN network framework, a packet encryption request refers to a series of subsequent packet encryption requests issued by an application to the controller through the northbound interface. It plays a crucial role in determining the security requirements of the packets. The request involves a significant volume of packets, indicating that allocating resources for the current network will pose greater challenges and requires the comprehensive consideration of the existing network resource status.

3.3. CPSR-HQ KDN Routing Scheme

Addressing the issues related to key management and routing in existing QKD networks, this paper proposes a Classified Packet Security Requirement-based QKD Network Routing scheme (CPSR-HQKDN). Considering the scenario where a large volume of packets with diverse security requirements arrive at backbone nodes within a short timeframe, CPSR-HQKDN adjusts the processing order of packets based on the included security requirement level information in the requests. This enables the fulfillment of a maximum number of key usage demands within the limited resources of the QKD network. Furthermore, CPSR-HQKDN grants a higher priority for retry attempts to packets that are prepared for discard due to timeout, aiming to enhance the overall service quality of the routing scheme.

Due to the classification based on the security requirements carried in the packet requests, the CPSR-HQKDN solution is capable of handling high-concurrency data transmission scenarios and significant gaps in packet security requirements compared to previous solutions. This section provides a detailed description of the packet security requirement classification, the processing queue flow structure, and the routing scheme. Table 1 presents the symbols used along with their meanings. It is important to note that the security requirement of a packet is not established by the packet itself, but rather assessed and defined by the routing based on the information carried in the request.

Table 1. Symbols and their meanings.

Symbols	Meanings
p_{sd}	Security requirement level for data packets, where N represents no use of QKD technology encryption, L represents the key generated using the BB84 protocol, and H represents the key generated using the MDI-QKD protocol.
T	Lifetime of the network topology
t	A single time slot, where $t \in T$
t_{last}	The previous time slot, where $t_{last} \in T$
p_{kd}	Key requirement for a single data packet
L	Set of all linkages in the network topology
l_t	Trusted relay link, where $l_t \in L$
l_{ut}	Untrusted relay link, where $l_{ut} \in L$
l_s	Current status of the link
N	Set of all nodes
n_{ut}	Untrusted relay node, where $n_{ut} \in N$
n_t	Trusted relay node, where $n_t \in N$
n_{back}	Backbone node, where $n_{back} \in N$
Q_n	Length of the queue of data packets awaiting encryption at node n
kr_l	Remaining key amount for the current link
kg_l	Key update rate for the current link
TD	Transmission delay for data packet transmission
p_n	Number of nodes a data packet goes through
$S(n)$	Amount of data packets waiting for forwarding at the current node

In the actual usage of quantum network packet transmission, the importance of key requests and specific encryption requirements varies. Purely employing key-based shortest path strategies or first-come-first-serve methods cannot provide sufficient service quality and high network resource utilization. In such scenarios, the CPSR-HQKDN scheme categorizes the packets to be forwarded based on the security requirement information included in the request headers sent by the application layer to the controller. Then, different routing schemes are determined for each category of packets.

First, a routing algorithm based on a transmission delay is employed for category N packets that do not require quantum key usage. It should be noted that in the QKD technology embedding scheme, the key preparation devices of the BB84 protocol and MDI-QKD protocol are added to the traditional SDN optical network. Therefore, when packets do not require QKD technology, they can also be forwarded using the traditional SDN optical network. Therefore, as shown in Figure 4, the diagram represents a hybrid network topology with two types of QKD links. It includes trusted relay nodes, untrusted relay nodes, and backbone nodes functioning as either senders or receivers. As shown in the figure, when backbone node 1 wants to forward category N packets to backbone node 5 via some intermediate nodes, there are two candidate paths: 1-2-3-4-5 and 1-6-7-8-5. The following is the routing weight target function for category N packets.

$$W = \alpha * \sum_{l \in L} \frac{TD_l}{2} + \beta * \sum_{n \in N} S(n), t = t_{last} \in T \quad (1)$$

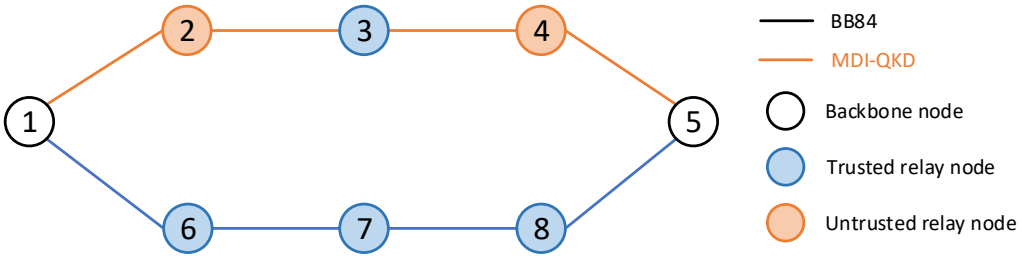

Figure 4. Analysis of Different Types of Packet Routing Cases.

In Equation (1), W represents the weight to be considered in the algorithm, which is determined by the transmission delay of the path in the previous time slot and the quantity of packets to be forwarded by the nodes along the path. α and β are two adjustment factors that must satisfy $\alpha + \beta = 1$, thus the weight W obtained should be minimized. The following Table 2 is the Dijkstra algorithm based on the minimum weight W.

The difference lies in the situation where, within a time slot t, when each port is processing packets requiring quantum keys, there is insufficient key availability, resulting in subsequent packets collectively waiting. This leads to a waste of time slot resources. In Figure 5, when such a situation occurs, it is allowed to set the waiting packets requiring quantum keys to the Blocked state and prioritize the forwarding of category N packets by allowing them to jump the queue.

Simultaneously, a key aspect of the proposed scheme involves the routing decision for category H and category L packets. As shown in Figure 5, when backbone node 1 needs to send a category H packet to backbone node 5 via intermediate nodes, its routing is restricted to using quantum keys generated by the MDI-QKD protocol. Therefore, the path can only be 1-2-3-4-5. Conversely, when backbone node 1 needs to send a category L packet to backbone node 5 via intermediate nodes, its routing is restricted to using quantum keys generated by the BB84 protocol. Hence, the path can only be 1-6-7-8-5. Under these conditions, the proposed scheme also takes into account key-related factors such as the key update rate, key consumption rate, key utilization rate, and remaining key quantity.

Table 2. Routing Algorithm for Class N Data Packets.

Input: t ∈ T, l ∈ L, n ∈ N, p, W Output: optimal path for a Class N Data Packet

Routing Algorithm for Class N Data Packets (topo, start, end):
weights = {vertex: infinity for vertex in topo}
weights[start] = 0
path = {}
visited = set()
while not all(weights[vertex] == infinity for vertex in topo):
current_vertex = min((vertex, weights[vertex]) for vertex in topo - visited)[0]
visited.add(current_vertex)
for neighbor, weight in topo[current_vertex].items():
if weights[neighbor] > weights[current_vertex] + w:
weights[neighbor] = weights[current_vertex] + w
path[neighbor] = current_vertex
shortest_path = []
current_vertex = end
while current_vertex != start:
shortest_path.append(current_vertex)
current_vertex = path[current_vertex]
shortest_path.append(start) shortest_path.reverse()
return shortest_path, weights[end]

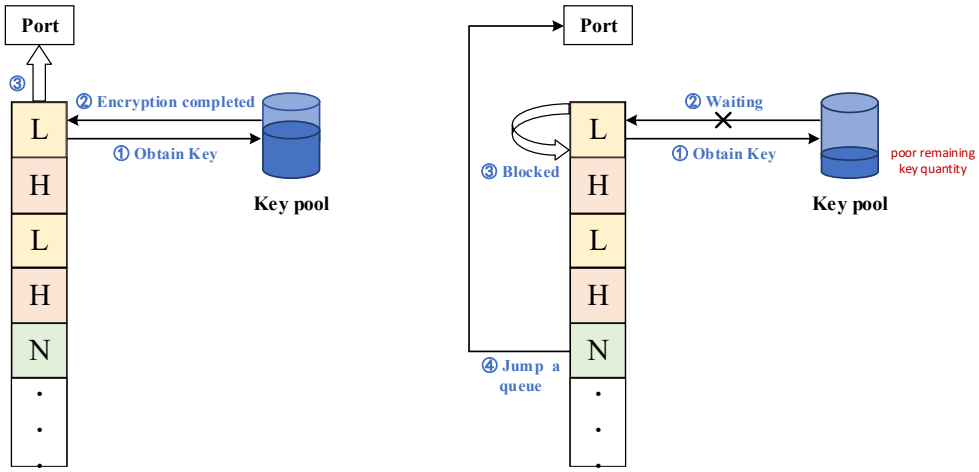

Figure 5. Schematic diagram of N-class packet queue insertion.

$$p_{kd} * (p_n - 1) \leq \sum_{l \in r} kr_l, r \in R \tag{2}$$

When a request to send an encrypted data packet is forwarded to the SDN controller, the controller needs to estimate whether a path exists in the network that satisfies the key consumption requirements of the data packet, such that it can reach the destination node. If there is no suitable path available, the request is rejected. Equation (2) represents the aforementioned logic, where R denotes the set of optional paths and r represents a single path in set R.

$$QBER_{l,t} = \left(1 - \frac{EB_{l,t}}{QB_{l,t}}\right) * 100\%, \forall t \in T, l \in L \tag{3}$$

$$SNR_{l,t} = \left(\frac{SP_{l,t}}{NP_{l,t}}\right) * 100\%, \forall t \in T, l \in L \tag{4}$$

$$QBER_{l,t} \geq \rho \tag{5}$$

$$SNR_{l,t} \geq \sigma \tag{6}$$

Due to external interference, such as noise or signal attenuation, the transmission of signals over the physical link, namely the optical fiber, may suffer from significant impairments. This can greatly impact the key update rate and packet transmission success rate. Thus, the controller needs to obtain the signal transmission quality information for both the quantum channel and the public channel in each time slot. Equations (3) and (4) are used to represent the signal transmission quality for the quantum channel and the public channel, respectively, where QBRR denotes the quantum bit transmission error rate, and SNR represents the signal-to-noise ratio. Equations (5) and (6) represent the thresholds below which the allocation of that particular link is abandoned until its signal quality recovers, with ρ and σ being the respective threshold values.

$$kr_l \geq p_{kd} \tag{7}$$

Equation (7) represents that when selecting the path for forwarding, it is also necessary to ensure that the remaining key quantity of the next-hop link is greater than the required key quantity for the data packet.

$$\omega = \delta * \frac{\sum_n Q_n}{p_n} + \varepsilon * \frac{p_n}{\sum_n kr_l}, n \in R_n \tag{8}$$

Based on these constraints, the remaining feasible paths are obtained and a final routing scheme is determined based on the remaining key quantity and hop count of each link. Equation (8) represents the calculation of the weights, which are determined by the length of the node port's waiting queue and the remaining key quantity. Here, R_n represents the set of nodes for a specific selectable path, and δ and ε are two adjustment factors. Table 3 illustrates the routing algorithm for L-class and H-class packets. Firstly, an initial weight matrix for the network topology needs to be constructed. Each element in the initial weight matrix corresponds to a tuple (l_c, l_s, ω, kr_l). It should be emphasized that the network topology differs between L-class and H-class packets, resulting in different weight matrices depending on the l_c condition. Then, according to Equations (5) and (6), it is determined whether the l_s in the current initial weight matrix meets the forwarding criteria. If it does not meet the criteria, it is set to 0. Subsequently, considering factors such as the number of waiting time slots carried by the packet and the required key quantity, an improved version of the Floyd algorithm is used to select the final path. Additionally, in this scheme, L-class and H-class packets that exceed the threshold of waiting time slots are directly discarded. In the source node, if the forwarded packet exceeds the expected response time, an opportunity for retransmission is provided to improve the quality of service.

Table 3. Routing Algorithm for Class L and H Data Packets.

Input: t ∈ T, l ∈ L, n ∈ N, p, $\boldsymbol{\omega}$ Output: optimal path for a Class L Data Packet or H Data Packet
Routing Algorithm for Class L and H Data Packets: Obtain the weight matrix X Check link status and update the weight matrix Get the packet waiting to be dequeued Get the waiting time slots of this packet If the waiting time slots are too long, just discard the packet directly Retrieve the required key amount for the packet, check if the remaining key amount in each link is sufficient, and update the different weight matrix according to $\boldsymbol{l_c}$ to contain only the variable $\boldsymbol{\omega}$ in the end

Table 3. *Cont.*

```
    for(int k = 1; k <= n; k++){
        for(int i = 1;I <= n; i++){
            if(k! = i){
                int t = (k < i)?W[i][k]: W[k][i];
                if(t == inf)continue;
                int temp = (k < i)?k:i;
                for(int j = 1; j <= temp; j++){
                    if(t + W[k][j] < W[i][j])W[i][j] = t + W[k][j];
                }
                for(int j = k + 1; j <= i; j++){
                    if(t + W[k][j] < W[i][j])W[i][j] = t + W[j][k];
                }
            }
        }
    }
    int start,end;
    if(start >= end)cout<<W[start][end]<<endl;
    else cout<<W[end][start]<<endl;
Obtain the optimal path based on the final W matrix
```

4. Network Simulation

As shown in Figure 6, to evaluate the performance of the CPSR-HQKDN routing scheme proposed in this paper, we conducted simulation experiments to compare the performance among different schemes. The network topology used for the experiment is shown in Figure 6. It consists of 17 nodes and 24 links. The weights of the links are marked in the diagram. Furthermore, for data packets with different security requirements, it is necessary to consider the links they can pass through. For example, L-class packets can only pass through the blue links generated using the BB84 protocol for quantum key generation, while H-class packets can only pass through the orange links generated using the MDI-QKD protocol for quantum key generation. On the other hand, N-class packets can pass through all links as they do not require the use of quantum keys.

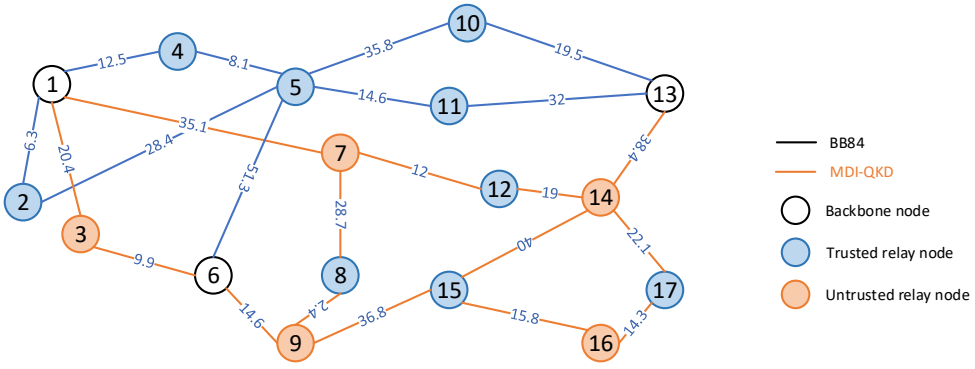

Figure 6. Topology diagram of simulation experiment.

The comparative schemes used in the simulation experiments are the APR-QKDN [35] scheme and the KL-SPRA [36] scheme. The CPSR-HQKDN routing scheme targets specific scenarios with significant differences in key requirements due to high concurrency and a large number of data packets with different security requirements. Therefore, it is necessary to reproduce this specific scenario when comparing the performance of each scheme. In each simulation experiment, the number of data packets to be forwarded is distributed in proportions of 50%, 30%, and 20% for the N-class, L-class, and H-class packets, respectively. Approximately 1000 data packets are forwarded at once, and the request time interval is continuously adjusted to observe the overall service quality and packet loss rate in the network environment.

As shown in Figure 7, the average packet loss rate for each class of data packets decreases as the data packet request time interval increases. When the data packet request time interval is sufficiently large, the number of keys stored in the node's key pool exceeds the total number of keys required by the data packets. It can be observed that when the data packet request interval reaches 250 ms, the packet loss rates for the three classes of data packets are very close to each other. This indicates that one of the main factors affecting the packet loss rate is the request time interval, which reflects the overall key consumption.

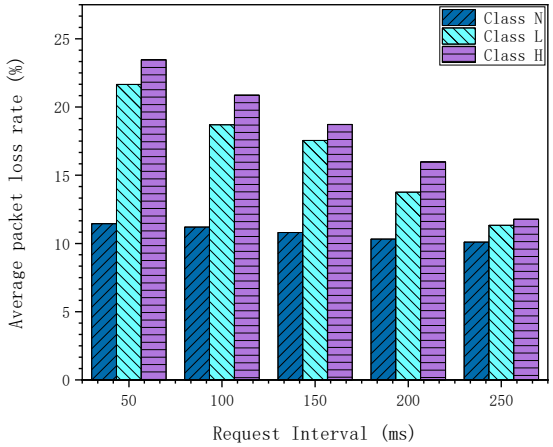

Figure 7. Relationship between average packet loss rate and packet request time interval.

By observing the average packet loss rate for N-class data packets, it can be seen that although it shows a decreasing trend, the fluctuation is not significant. This is because of the pre-emptive mechanism provided by CPSR-HQKDN, which prevents N-class data packets from being stuck in the waiting queue due to the lack of quantum keys.

As shown in Figure 8, the average packet loss rate for CPSR-HQKDN and the other schemes decreases as the data packet request time interval increases. This is because the local storage of remaining quantum keys at the quantum nodes increases with the growing data packet request time interval, making network resources gradually abundant to serve more requests. Meanwhile, CPSR-HQKDN exhibits a significant performance improvement compared to APR-QKDN and KL-SPRA in terms of the packet loss rate. The performance difference between CPSR-HQKDN and APR-QKDN remains around 9%, while the difference between CPSR-HQKDN and KL-SPRA is around 16%. This is attributed to the appropriate path cost calculation strategy adopted by CPSR-HQKDN, along with the provision of additional delay-based retransmission opportunities for those data packets requiring quantum keys, which reduces the network packet loss rate and improves network service quality.

As depicted in Figure 9, the success rate of CPSR-HQKDN, APR-QKDN, and KL-SPRA requests varies with the request interval, with CPSR-HQKDN exhibiting better performance than the other schemes as the request interval increases. Compared to APR-QKDN, CPSR-HQKDN has an approximately 3% higher success rate, while a 7% improvement is observed compared to KL-SPRA. Additionally, it can be observed from experimental results that the success rate of L-class data packet requests is slightly higher than that of H-class data packets, which reflects the faster key updating rate of the BB84 protocol as compared to the MDI protocol. When the request interval exceeds 200 ms, the performance of APR approaches that of CPSR-HQKDN, which is mainly due to the device foundation built by the idle period-supplemented key pool construction. However, this also incurs greater deployment costs and increased facility construction pressure.

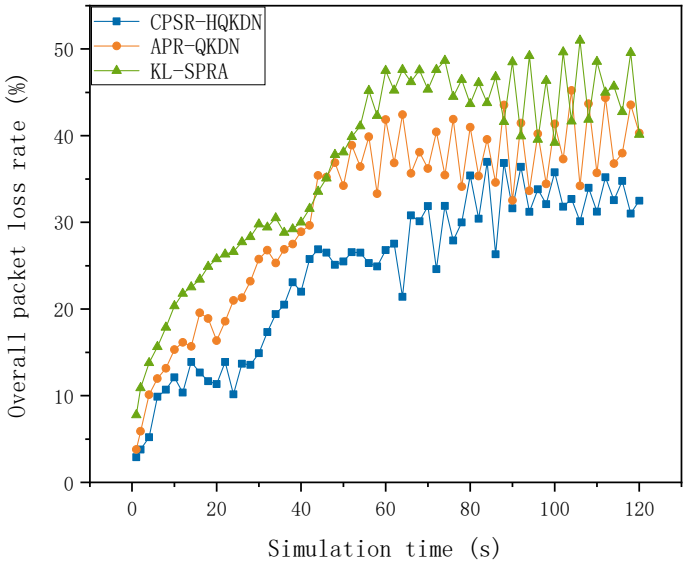

Figure 8. Overall packet loss rate of the network as a function of simulation time.

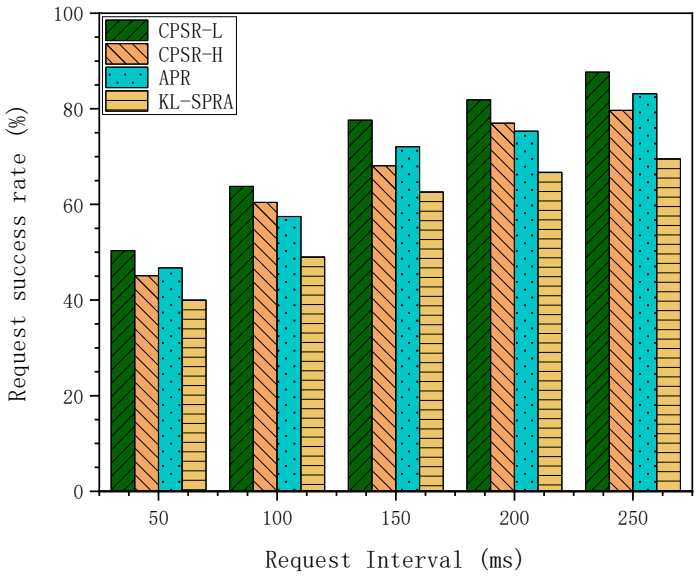

Figure 9. Relationship between packet request success rate and request time interval.

As shown in Figure 10, the success rate of all schemes exhibits a decreasing trend followed by fluctuations during the dynamic simulation time. This is because in the simulated environment, the key pool is first filled with a sufficient number of keys before sending out data packet requests, which prolongs the overall network lifespan. However, there is a certain gap between the capacity of the key pool and the total key demand of data packets at a request interval of 50 ms, resulting in an inability to fulfill key replenishment within the request interval and leading to a decrease in the success rate. As the simulation time continues to increase, the success rate gradually fluctuates within a certain range. In terms of the success rate, CPSR-HQKDN achieves at least a 16% performance improvement compared to the APR-QKDN and KL-SPRA schemes.

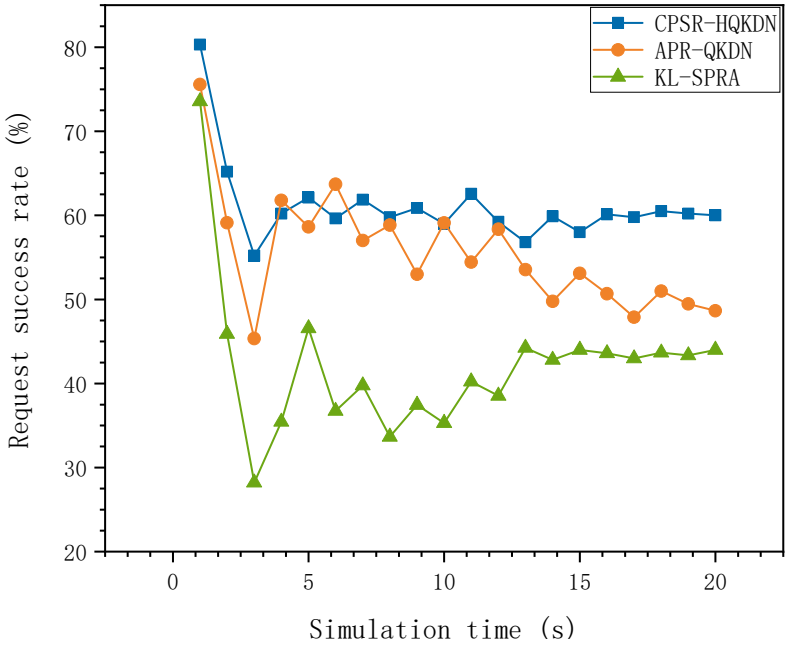

Figure 10. Change in request success rate with simulation time.

As shown in Figure 11, the key utilization rates of CPSR-HQKDN, APR-QKDN, and KL-SPRA significantly increase with the simulation time, and their trends are similar. This is because, initially, the amount of keys used is not sufficient to reach a stable standard, leading to a decrease in the key utilization rate due to packet loss. The CPSR-HQKDN scheme has several advantages over the other schemes in terms of the key utilization rate. It offers an approximately 7% improvement compared to APR-QKDN and around a 15% improvement compared to KL-SPRA.

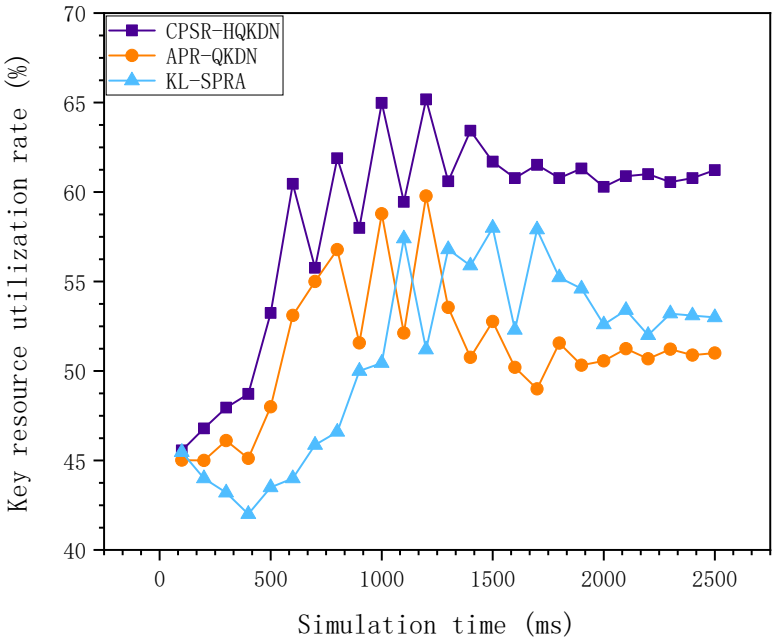

Figure 11. Change in Key Utilization with Simulation Time.

5. Conclusions and Outlook

This paper primarily introduces a routing scheme based on the classification of packet security requirements in a hybrid Quantum Key Distribution (QKD) network within the Software-Defined Networking (SDN) framework. In existing QKD routing schemes,

typically, only one type of QKD link is utilized for the network design, or the routing algorithm solely considers the length of the path or the quality of service, overlooking the distinct characteristics of different packet security levels. In contrast, the proposed scheme employs a hybrid QKD network model composed of two types of links: the BB84 protocol and the MDI-QKD protocol. Additionally, this scheme establishes a unique linear programming mathematical model based on the key bit generation features of BB84 and MDI-QKD, and proposes a routing scheme based on the classification of packet security requirements. Furthermore, it incorporates a new theory of queuing encrypted packets to alleviate congestion, reduce the packet loss rate, and enhance quantum key utilization in the overall quantum network.

Firstly, the data packets are classified into different security levels according to their sensitivity and confidentiality requirements. The level information is then added to the request header of the packet. Subsequently, the packets are classified based on the request header of the encryption request during the routing process. During routing, the packets are forwarded through paths corresponding to the appropriate security levels, ensuring the security of the packet transmission. At the same time, the slow generation speed, high cost, and precious nature of the keys generated by QKD technology are taken into account to improve the key utilization rate and the overall quality of service in the network environment. The experimental results demonstrate that the routing scheme based on the classification of packet security requirements effectively reduces the consumption of quantum keys and improves the overall network service quality. It can select suitable paths for transmission based on the demands of packets with different security levels, thereby reducing the risks of eavesdropping, tampering, or delayed packet loss.

However, there are still potential areas for improvement in this routing scheme. On one hand, further research can be conducted on how to conceal request header information in the design of routing protocols, and consider introducing traditional keys to confound the security levels, optimizing the overall network security. On the other hand, although this routing scheme is based on emerging architectures such as Software-Defined Networking (SDN), insufficient integration has been made to enhance the overall network service quality and security. Future exploration can be conducted on the integration of packet security requirement classification with novel network architectures. In conclusion, the routing scheme based on the classification of packet security requirements holds potential and promising applications in enhancing the security of network packets. With further research and improvements, this scheme can be made more effective and feasible in practical network environments.

Author Contributions: L.B. and W.W. played a key role in the initial conception of the ideas. W.W. and X.Y. further developed and refined these ideas. M.M. and X.D. were responsible for writing the source code and conducting the simulations. X.D. and Z.J. provided valuable input during the editing process. All authors actively contributed to writing the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Natural Science Foundation of the Jilin Province (Grant No. 20210101417JC).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to internal use in laboratory.

Acknowledgments: The authors would like to thank the reviewers for their valuable comments and suggestions.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ladd, T.D.; Jelezko, F.; Laflamme, R.; Nakamura, Y.; Monroe, C.; O'Brien, J.L. Quantum computers. *Nature* **2010**, *464*, 45–53. [CrossRef] [PubMed]
2. Arute, F.; Arya, K.; Babbush, R.; Bacon, D.; Bardin, J.C.; Barends, R.; Biswas, R.; Boixo, S.; Brandao, F.G.S.L.; Buell, D.A.; et al. Quantum supremacy using a programmable superconducting processor. *Nature* **2019**, *574*, 505–510. [CrossRef] [PubMed]
3. Milanov, E. The RSA algorithm. *RSA Lab.* **2009**, 1–11.
4. Shor, P.W. Algorithms for quantum computation: Discrete logarithms and factoring. In Proceedings of the 35th Annual Symposium on Foundations of Computer Science, Santa Fe, NM, USA, 20–22 November 1994; pp. 124–134.
5. Shor, P.W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Rev.* **1999**, *41*, 303–332. [CrossRef]
6. Mallouli, F.; Hellal, A.; Saeed, N.S.; Alzahrani, F.A. A survey on cryptography: Comparative study between RSA vs ECC algorithms, and RSA vs. El-Gamal algorithms. In Proceedings of the 2019 6th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2019 5th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom), Paris, France, 21–23 June 2019; pp. 173–176.
7. Grover, L.K. A fast quantum mechanical algorithm for database search. In Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing, Philadelphia, PA, USA, 22–24 May 1996; pp. 212–219.
8. Diamanti, E.; Lo, H.-K.; Qi, B.; Yuan, Z. Practical challenges in quantum key distribution. *NPJ Quantum Inf.* **2016**, *2*, 16025. [CrossRef]
9. Lo, H.-K.; Curty, M.; Tamaki, K. Secure quantum key distribution. *Nat. Photonics* **2014**, *8*, 595–604. [CrossRef]
10. Ahn, J.; Kwon, H.-Y.; Ahn, B.; Park, K.; Kim, T.; Lee, M.-K.; Kim, J.; Chung, J. Toward quantum secured distributed energy resources: Adoption of post-quantum cryptography (pqc) and quantum key distribution (qkd). *Energies* **2022**, *15*, 714. [CrossRef]
11. Ahmad, I.; Shahabuddin, S.; Kumar, T.; Okwuibe, J.; Gurtov, A.; Ylianttila, M. Security for 5G and beyond. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 3682–3722. [CrossRef]
12. Zhang, Q.; Xu, F.; Chen, Y.A.; Peng, C.Z.; Pan, J.W. Large scale quantum key distribution: Challenges and solutions [Invited]. *Opt. Express* **2018**, *26*, 24260–24273. [CrossRef]
13. Nawaz, S.J.; Sharma, S.K.; Wyne, S.; Patwary, M.N.; Asaduzzaman, M. Quantum machine learning for 6G communication networks: State-of-the-art and vision for the future. *IEEE Access* **2019**, *7*, 46317–46350. [CrossRef]
14. Chong, S.-K.; Hwang, T. Quantum key agreement protocol based on BB84. *Opt. Commun.* **2010**, *283*, 1192–1195. [CrossRef]
15. Lo, H.-K.; Curty, M.; Qi, B. Measurement-device-independent quantum key distribution. *Phys. Rev. Lett.* **2012**, *108*, 130503. [CrossRef] [PubMed]
16. Liu, Y.; Chen, T.-Y.; Wang, L.-J.; Liang, H.; Shentu, G.-L.; Wang, J.; Cui, K.; Yin, H.-L.; Liu, N.-L.; Li, L. Experimental measurement-device-independent quantum key distribution. *Phys. Rev. Lett.* **2013**, *111*, 130502. [CrossRef]
17. Cao, Y.; Zhao, Y.; Wang, Q.; Zhang, J.; Ng, S.X.; Hanzo, L. The evolution of quantum key distribution networks: On the road to the qinternet. *IEEE Commun. Surv. Tutor.* **2022**, *24*, 839–894. [CrossRef]
18. Wang, S.; Yin, Z.-Q.; He, D.-Y.; Chen, W.; Wang, R.-Q.; Ye, P.; Zhou, Y.; Fan-Yuan, G.-J.; Wang, F.-X.; Chen, W. Twin-field quantum key distribution over 830-km fibre. *Nat. Photonics* **2022**, *16*, 154–161. [CrossRef]
19. Zhang, W.; van Leent, T.; Redeker, K.; Garthoff, R.; Schwonnek, R.; Fertig, F.; Eppelt, S.; Rosenfeld, W.; Scarani, V.; Lim, C.C.-W. A device-independent quantum key distribution system for distant users. *Nature* **2022**, *607*, 687–691. [CrossRef]
20. Pompili, M.; Hermans, S.L.; Baier, S.; Beukers, H.K.; Humphreys, P.C.; Schouten, R.N.; Vermeulen, R.F.; Tiggelman, M.J.; dos Santos Martins, L.; Dirkse, B. Realization of a multinode quantum network of remote solid-state qubits. *Science* **2021**, *372*, 259–264. [CrossRef]
21. Qi, Z.; Li, Y.; Huang, Y.; Feng, J.; Zheng, Y.; Chen, X. A 15-user quantum secure direct communication network. *Light Sci. Appl.* **2021**, *10*, 183. [CrossRef]
22. Wengerowsky, S.; Joshi, S.K.; Steinlechner, F.; Hübel, H.; Ursin, R. An entanglement-based wavelength-multiplexed quantum communication network. *Nature* **2018**, *564*, 225–228. [CrossRef]
23. Cao, Y.; Zhao, Y.; Li, J.; Lin, R.; Zhang, J.; Chen, J. Hybrid Trusted/Untrusted Relay-Based Quantum Key Distribution over Optical Backbone Networks. *IEEE J. Sel. Areas Commun.* **2021**, *39*, 2701–2718. [CrossRef]
24. Xu, F.; Zhang, Y.Z.; Zhang, Q.; Pan, J.W. Device-independent quantum key distribution with random postselection. *Phys. Rev. Lett.* **2022**, *128*, 110506. [CrossRef]
25. Ma, W.; Chen, B.; Liu, L.; Chen, H.; Shao, W.; Gao, M.; Wu, J.; Ho, P.-H. Equilibrium Allocation Approaches of Quantum Key Resources with Security Levels in QKD-Enabled Optical Data Center Networks. *IEEE Internet Things J.* **2022**, *9*, 25660–25672. [CrossRef]
26. Yu, X.; Liu, Y.; Zou, X.; Cao, Y.; Zhao, Y.; Nag, A.; Zhang, J. Secret-Key Provisioning With Collaborative Routing in Partially-Trusted-Relay-based Quantum-Key-Distribution-Secured Optical Networks. *J. Light. Technol.* **2022**, *40*, 3530–3545. [CrossRef]
27. Wang, C.; Kon, W.Y.; Ng, H.J.; Lim, C.C. Experimental symmetric private information retrieval with measurement-device-independent quantum network. *Light Sci. Appl.* **2022**, *11*, 268. [CrossRef] [PubMed]
28. Luo, W.; Cao, L.; Shi, Y.; Wan, L.; Zhang, H.; Li, S.; Chen, G.; Li, Y.; Li, S.; Wang, Y.; et al. Recent progress in quantum photonic chips for quantum communication and internet. *Light Sci. Appl.* **2023**, *12*, 175. [CrossRef] [PubMed]

29. Wang, S. Symmetric private information retrieval supported by quantum-secure key-exchange network. *Light Sci. Appl.* **2022**, *11*, 301. [CrossRef] [PubMed]
30. Ren, S.; Wang, Y.; Su, X. Hybrid quantum key distribution network. *Sci. China Inf. Sci.* **2022**, *65*, 200502. [CrossRef]
31. Liu, R.; Rozenman, G.G.; Kundu, N.K.; Chandra, D.; De, D. Towards the industrialisation of quantum key distribution in communication networks: A short survey. *IET Quantum Commun.* **2022**, *3*, 151–163. [CrossRef]
32. Nauerth, S.; Fürst, M.; Schmitt-Manderbach, T.; Weier, H.; Weinfurter, H. Information leakage via side channels in freespace BB84 quantum cryptography. *New J. Phys.* **2009**, *11*, 065001. [CrossRef]
33. Bloom, Y.; Fields, I.; Maslennikov, A.; Rozenman, G.G. Quantum cryptography—A simplified undergraduate experiment and simulation. *Physics* **2022**, *4*, 104–123. [CrossRef]
34. Wang, N.; Tian, X.; Zhang, X.; Lin, S. Quantum Secure Multi-Party Summation with Identity Authentication Based on Commutative Encryption. *Photonics* **2023**, *10*, 558. [CrossRef]
35. Chen, L.; Zhang, Z.; Zhao, M.; Yu, K.; Liu, S. APR-QKDN: A Quantum Key Distribution Network Routing Scheme Based on Application Priority Ranking. *Entropy* **2022**, *24*, 1519. [CrossRef] [PubMed]
36. Yang, C.; Zhang, H.; Su, J. The qkd network: Model and routing scheme. *J. Mod. Opt.* **2017**, *64*, 2350–2362. [CrossRef]