

A new quantum key distribution resource allocation androuting optimization scheme

Lin Bi(毕琳), Xiaotong Yuan(袁晓同), Weijie Wu(吴炜杰), and Shengxi Lin(林升熙)

Citation: Chin. Phys. B, 2024, 33 (**3**): 030309. DOI: 10.1088/1674-1056/ad2503
Journal homepage: http://cpb.iphy.ac.cn; http://iopscience.iop.org/cpb

What follows is a list of articles you may be interested in

Improved decoy-state quantum key distribution with uncharacterized heralded single-photon sources

Le-Chen Xu(徐乐辰), Chun-Hui Zhang(张春辉), Xing-Yu Zhou(周星宇), and Qin Wang(王琴)
Chin. Phys. B, 2024, 33 (**2**): 020313. DOI: 10.1088/1674-1056/ad062a

Effect of weak randomness flaws on security evaluation of practical quantum key distribution with distinguishable decoy states

Yu Zhou(周雨), Hong-Wei Li(李宏伟), Chun Zhou(周淳), Yang Wang(汪洋), Yi-Fei Lu(陆宜飞),Mu-Sheng Jiang(江木生), Xiao-Xu Zhang(张晓旭), and Wan-Su Bao(鲍皖苏)
Chin. Phys. B, 2023, 32 (**5**): 050305. DOI: 10.1088/1674-1056/ac8730

Security of the traditional quantum key distribution protocols with finite-key lengths

Bao Feng(冯宝), Hai-Dong Huang(黄海东), Yu-Xiang Bian(卞宇翔), Wei Jia(贾玮), Xing-Yu Zhou(周星宇), and Qin Wang(王琴)
Chin. Phys. B, 2023, 32 (**3**): 030307. DOI: 10.1088/1674-1056/ac9b33

Performance of phase-matching quantum key distribution based on wavelength division multiplexing technology

Haiqiang Ma(马海强), Yanxin Han(韩雁鑫), Tianqi Dou(窦天琦), and Pengyun Li(李鹏云)
Chin. Phys. B, 2023, 32 (**2**): 020304. DOI: 10.1088/1674-1056/ac6ee3

Temperature characterizations of silica asymmetric Mach-Zehnder interferometer chip for quantum key distribution

Dan Wu(吴丹), Xiao Li(李骁), Liang-Liang Wang(王亮亮), Jia-Shun Zhang(张家顺), Wei Chen(陈巍), Yue Wang(王玥), Hong-Jie Wang(王红杰), Jian-Guang Li(李建光), Xiao-Jie Yin(尹小杰), Yuan-Da Wu(吴远大), Jun-Ming An(安俊明), and Ze-Guo Song(宋泽国)
Chin. Phys. B, 2023, 32 (**1**): 010305. DOI: 10.1088/1674-1056/ac9224

A new quantum key distribution resource allocation and routing optimization scheme

Lin Bi(毕琳)[1,2], Xiaotong Yuan(袁晓同)[1,2,†], Weijie Wu(吴炜杰)[1,2], and Shengxi Lin(林升熙)[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*

(Received 12 October 2023; revised manuscript received 24 January 2024; accepted manuscript online 2 February 2024)

Quantum key distribution (QKD) is a technology that can resist the threat of quantum computers to existing conventional cryptographic protocols. However, due to the stringent requirements of the quantum key generation environment, the generated quantum keys are considered valuable, and the slow key generation rate conflicts with the high-speed data transmission in traditional optical networks. In this paper, for the QKD network with a trusted relay, which is mainly based on point-to-point quantum keys and has complex changes in network resources, we aim to allocate resources reasonably for data packet distribution. Firstly, we formulate a linear programming constraint model for the key resource allocation (KRA) problem based on the time-slot scheduling. Secondly, we propose a new scheduling scheme based on the graded key security requirements (GKSR) and a new micro-log key storage algorithm for effective storage and management of key resources. Finally, we propose a key resource consumption (KRC) routing optimization algorithm to properly allocate time slots, routes, and key resources. Simulation results show that the proposed scheme significantly improves the key distribution success rate and key resource utilization rate, among others.

Keywords: quantum key distribution (QKD), resource allocation, key storage, routing algorithm

PACS: 03.67.Dd, 03.67.Hk, 42.82.Ds, 64.60.aq **DOI:** 10.1088/1674-1056/ad2503

1. Introduction

With the increasing demand for security in network communications, the introduction of quantum communication technology and the construction of quantum networks are crucial for ensuring high-security data transmission in future network systems. Quantum key distribution (QKD) technology is an essential component of quantum communication, providing a secure key distribution mechanism for quantum networks. There are significant differences between QKD networks and classical networks in terms of security principles and implementation methods.

Classical key distribution in network systems is typically based on mathematical problems such as large number factorization or discrete logarithm problems. The security of these key systems relies on the current computational capabilities being unable to solve these mathematical problems within practical timeframes. However, with the continuous development of quantum computers, these systems may be vulnerable to being broken within practical timeframes using algorithms such as Grover's algorithm or Shor's algorithm.[1,2] On the other hand, the key distribution process in QKD networks does not rely on the complexity of computations, but rather utilizes the fundamental principles of quantum mechanics, namely the Heisenberg uncertainty principle and the quantum no-cloning theorem.[3–5] In theory, QKD can achieve absolute security, particularly against threats such as man-in-the-middle attacks and the rapid development of computational power.

Furthermore, the encryption methods used in classical networks typically rely on public key encryption systems, where the same key is used for encryption during end-to-end transmission. In contrast, QKD networks operate differently. Taking the trusted relay link based on the BB84 protocol as an example, quantum keys are generated using quantum bits (e.g., polarized photons) and shared between the sender (receiver) and adjacent relay nodes. The encryption, decryption, and destruction of the ciphertext all occur between point-to-point connections. Therefore, compared to the end-to-end public keys used in classical networks, the quantum keys in QKD networks have shorter lifecycles and higher security performance.[6]

When data packets are encrypted using quantum keys, the efficiency of encryption can be effectively improved and key consumption can be reduced by utilizing the advanced encryption standard (AES) algorithm. Additionally, by constructing a quantum key pool, quantum key resources can be accumulated to provide a stable and timely flow of quantum keys to various nodes in the network, thereby enhancing the capacity of QKD networks to handle encryption service requests. Currently, most QKD networks can only achieve a key generation rate of 1.2 Mbps on a 50.5 km fiber link and a key generation rate of 6.5 bps on a 405 km fiber link.[7,8] Quantum keys are crucial resources in QKD networks, and given the massive data flow in today's networks, the key generation rates fall far short of the requirements for data encryption. Therefore, the

[†]Corresponding author. E-mail: yxt8090@163.com

© 2024 Chinese Physical Society and IOP Publishing Ltd

construction of a quantum key pool is of great significance for the development of QKD networks.

Due to the complexity of resource variations, such as keys, on different links, it is challenging to allocate these resources effectively to data packets during transmission. As a result, the key utilization rate, overall service quality, and even network lifespan are greatly impacted. Additionally, when multiple nodes simultaneously request keys, key request conflicts may occur, leading to request delays or rejections. Although quantum key distribution is theoretically secure, there are potential security risks in practical storage and management, such as unauthorized access, malicious tampering, and leakage.[9,10] Therefore, further research is needed on how to securely backup and update quantum keys, while ensuring the integrity and confidentiality of backup data. Additionally, with the integration of QKD and existing communication systems, there are still issues to be addressed, such as security authentication and system integration.

In QKD networks, the efficiency and success rate of routing are also important issues that deserve research attention. By selecting the optimal path, data can be transmitted efficiently and securely, reducing network latency and key consumption, improving key distribution efficiency, and optimizing the allocation of network resources.[11] Additionally, considering the scalability and robustness of the network, optimizing routing strategies can enhance the network's scalability and adaptability to dynamic environments and node failures. Therefore, researching routing strategies is crucial for improving the reliability of QKD networks and ensuring the fair allocation of resources.

To address the aforementioned issues, this paper proposes a novel QKD resource allocation and routing optimization scheme, with the following key contributions.

(i) Firstly, we formulate a maximum successful key service request model and a minimum key consumption model for the key resource allocation (KRA) problem, in preparation for subsequent transmission path optimization schemes.

(ii) Next, a key resource scheduling scheme based on the graded key security requirements (GKSR) and a key storage algorithm based on the micro-log are proposed, enabling effective storage and management of key resources.

(iii) Finally, a key resource consumption (KRC) routing optimization algorithm is introduced. Performance of the proposed algorithm is simulated in two typical network topologies through simulation experiments. The simulation results demonstrate the superiority of this routing scheme.

The remaining sections of this paper are organized as follows. Section 2 introduces the relevant concepts and technologies, including point-to-point QKD protocols, trusted relay QKD network principles, and SDN-based QKD network architectures. Section 3 provides detailed explanations of the linear programming constrained model and the KRC routing algorithm. Section 4 analyzes the results of simulation experiments. Finally, Section 5 presents the summary and conclusions.

2. Related concepts and technologies

2.1. Related work

As quantum key distribution (QKD) becomes the preferred method for optical network information encryption, the lifespan of keys between point-to-point connections varies due to the different key generation rates and inventories on each link. This can lead to various issues. Firstly, the selection of paths has a significant impact on key quantity changes. Insufficient key quantity on a link will inevitably result in data packets waiting at nodes, increasing latency and potentially leading to congestion, greatly reducing the quality of network services. Secondly, considering the low generation rate of quantum keys, it is necessary to deploy multiple key generation devices at a single node. This raises issues regarding deployment costs and how to allocate device resources effectively. Thirdly, the entire QKD network system needs to differentiate between quantum channels, public channels, and data transmission channels. To improve the generation rate of quantum keys, channel multiplexing techniques inevitably need to be employed, which involve the fair allocation of channel resources. Finding intermediate points to address the routing, channel, and device resource allocation issues in a coordinated manner is a research-worthy problem.

In Ref. [12] an advanced QKD secured optical network architecture is proposed, introducing the concept of a quantum key pool (QKP) to effectively manage key resources, enabling efficient key distribution and utilization. Additionally, Ref. [13] introduces a key-as-a-service (KaaS) QKD secured optical network architecture, collectively addressing the efficient deployment and utilization of keys. Furthermore, in Ref. [14] a quantum-as-a-service (QaaS) SDN new architecture, referred to as the SDQaaS framework, achieves QaaS for multiple users through the QKD network infrastructure. QaaS allows multiple users to request different quantum keys from the same network infrastructure to obtain the desired key rates.[15] In Ref. [16] the routing, wavelength, and time slot allocation (RWTA) problem is investigated, and a solution applicable to static traffic scenarios is proposed. This approach is based on an integer linear programming (ILP) model and introduces heuristic algorithms to address resource allocation issues. For dynamic traffic scenarios, time conflicts arise due to LP/QLP requests arriving simultaneously in the network. To overcome this challenge, Ref. [17] introduces the concept of a time sliding window (TSW). However, in QKD secured optical networks, a balance must be struck between security levels

and resource utilization. To achieve this balance, Ref. [18] presents an on-demand key (KoD) strategy based on the QKP construction technology, ensuring the security of control channels (CCh) and data channels (DCh) on software defined optical networks (SDON).

Currently, several countries both domestically and internationally have established trusted-node quantum key distribution (QKD) networks.[19] However, numerous challenges persist within QKD networks that require resolution. These challenges encompass aspects such as optimizing system parameters in QKD network configuration,[20] optimizing system stability control,[21] and selecting suitable routing schemes for QKD networks. Investigating the routing mechanisms in QKD networks holds significant importance for their applications and development. Some routing schemes in QKD networks consider variations in key requests, making them more suitable for practical applications. For instance, Cao *et al.*[16] initially proposed a scheme for allocating keys based on security requirements, defining different security scenarios. However, their research lacks mechanisms to prevent request blocking in specific scenarios. Ma *et al.*[22] presented two routing, wavelength, and time slot allocation (RWTA) schemes, flexible security level (FSL) and specific security level (SSL). RWTA–FSL lowers security levels to facilitate more successful link requests and demonstrates strong adaptability. Nevertheless, this approach still has limitations in high-concurrency scenarios. Chen[23] introduced an adaptive QKD routing scheme based on application requirements, significantly improving request success rates but without specific scenario analysis. Yu *et al.*[24] proposed a heuristic collaborative routing algorithm for partially trusted relay QKD networks, considering relay deployment and remaining key quantities, but with certain deficiencies in handling request disparities. Additionally, some routing schemes[25–30] prioritize subsequent processing requests based on criteria such as security levels, business variations, and request arrival times. These schemes can accommodate diverse domain requirements with relatively few restrictions concerning efficiency enhancement considerations.

The aforementioned research lacks consideration for scenarios with multi-level data security requirements and scarce key resources. Therefore, in this paper, we address such special scenarios by formulating a linear programming constraint model and proposing the KRC routing algorithm for key resource allocation (KRA) based on time slot assignment. We also propose the GKSR scheduling scheme and a novel micro-log key storage algorithm to effectively store and manage key resources.

2.2. Peer to peer QKD protocol

Most QKD protocols utilize quantum superposition states or quantum entanglement effects to achieve quantum state information transmission or key distribution, theoretically guaranteeing information-theoretic security.[31] QKD channels typically consist of both quantum channels and classical channels. The quantum channel is used to transmit quantum states encoding classical information, while the classical channel is employed to transmit classical information for synchronization and key negotiation between the sender and receiver.[32]

If an eavesdropper intercepts a portion of quantum states from the quantum channel, those quantum states will collapse. In this case, both parties will detect the presence of the eavesdropper through the measurement results of the measurement bases. Consequently, the bits that were supposed to become the key will be discarded. Additionally, the eavesdropper may attempt to measure and replicate these quantum states before sending them to the receiver. However, according to the quantum no-cloning theorem, the cloned quantum states will inevitably collapse, resulting in noticeable errors in the measurement results. Therefore, any potential eavesdropping activity can be detected.

QKD sending and receiving ends connected via a QKD channel follow specific QKD protocols, executing a series of processes to establish a shared symmetric key. These processes ensure secure key distribution and protect the key's security by detecting any potential eavesdropping activity.

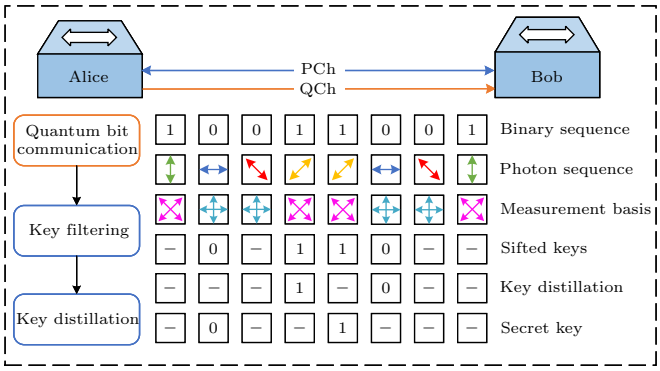


Fig. 1. Point-to-point QKD based on BB84 protocol.

The first QKD protocol, known as BB84, was proposed by Bennett and Brassard in 1984, and its security has been proven by researchers. Currently, the BB84 protocol continues to be widely used in commercial QKD systems. As illustrated in Fig. 1, point-to-point QKD implemented by BB84 can consist of the following phases.

(i) Initialization: Alice and Bob prearrange the encoding scheme for the quantum bits to be used. They select two orthogonal bases as encoding bases, typically horizontal and vertical polarization states (e.g., 0° and 90°) as well as diagonal and antidiagonal polarization states (e.g., 45° and 135°).

(ii) Generation and transmission of quantum bits: Alice randomly selects the bits to be sent and encodes them into the

corresponding quantum states using the predetermined encoding scheme. She transmits the quantum bits to Bob by sending photons in the polarization state (e.g., using polarization filters).

(iii) Basis selection: after receiving the quantum bits sent by Alice, Bob randomly selects a measurement basis (e.g., horizontal/vertical basis or diagonal/antidiagonal basis) for measurement.

(iv) Public disclosure: Alice and Bob publicly reveal their respective choices of encoding bases, measurement bases, and measurement outcomes. They can compare this publicly disclosed information to determine whether the results measured in the same basis are consistent.

(v) Error correction: Alice and Bob first declare in which measurements they used the same bases. Then, they compare the measurement outcomes for these shared bases. If discrepancies are found, it indicates potential eavesdropping or errors due to channel noise. They exclude these measurement outcomes from the key.

(vi) Key extraction: Alice and Bob utilize the measurement outcomes retained based on publicly disclosed results, where the measurement outcomes in shared bases are consistent, as the final key bits. These key bits are error-corrected, and under the assumption of no eavesdropping, the key is considered secure.

(vii) Security verification: to verify the security of the generated key, Alice and Bob can choose a small subset of key bits for inspection. They publicly disclose these bits and compare them for consistency. If they match, it indicates that the generated key is secure.

The crucial aspect of the BB84 protocol resides in exploiting the inherent un-clonability of quantum states and the unavoidable presence of eavesdropping. This is achieved by employing public comparison and error correction mechanisms to guarantee the security of the final key. Furthermore, the verification process serves to provide additional confirmation of the key's security.

2.3. Principles of trusted relay QKD network

QKD networks can be categorized into three types: optical-node-based QKD networks, quantum-relay-based QKD networks, and trusted-relay-based QKD networks.[33–35] Under the condition of quantum key distribution (QKD) optical networks, trusted relays are currently the most effective method for achieving long-distance transmission and enabling large-scale network architectures. QKD networks based on trusted relays exhibit higher security and scalability, and have been applied in various practical QKD networks. For example, the first QKD network deployed in the United States, DARPA,[36] and the SECOQC network in the European Union,[37] both use trusted relays for long-distance fiber transmission. The feasibility of trusted relays in QKD networks has been validated.[38]

The principle of a point-to-point QKD link based on trusted relays is illustrated in Fig. 2. An intermediate node (QKD node2) equipped with a trusted relay is placed between the source node (QKD node1) and the destination node (QKD node3). Each pair of directly connected QKD nodes continuously generates keys and stores them in the QKD key pool (QKP).

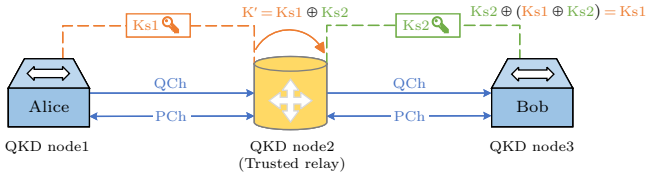


Fig. 2. Example of point-to-point QKD link based on trusted relay.

Initially, Alice establishes an initial key (Ks1) with QKD node2 through the BB84 protocol. QKD node2 also establishes a key (Ks2) with Bob, which is of the same size as Ks1. Upon receiving the QKD request, QKD node2 employs the XOR algorithm to generate $K' = Ks1 \oplus Ks2$, which is then transmitted to Bob. Subsequently, Bob decrypts $Ks2 \oplus (Ks1 \oplus Ks2) = Ks1$ to retrieve Ks1, thereby enabling the sharing of the identical key with Alice. In the context of QKD based on trusted relays, the key consumption is represented by $Ks1 + Ks2$. This consumption encompasses the shared key (Ks1) between Alice and Bob and the key (Ks2) utilized for encryption.

Due to the utilization of key distribution methods that provide unconditional security, the keys shared between adjacent relay nodes during key transmission remain confidential. Additionally, the trusted nature of the relay nodes ensures the theoretical security of the key delivery mechanism based on trusted relays.

2.4. SDN based QKD network architecture

In this section, the KRA problem is applied to a real QKD network, and a centralized software defined network (SDN) controller is used, as shown in Fig. 3. This model consists of four layers: the application layer, control layer, key management layer, and QKD layer. The northbound interface protocol (such as RESTful API[39]) is used for communication between the application layer and control layer, while the southbound interface protocol (such as OpenFlow[40] and NETCONF[41]) is used for communication between the control layer and QKD layer. OpenFlow is a highly potential SDN protocol that allows remote management of the forwarding plane using the controller, while NETCONF is a transaction-based protocol that provides access and modification functions for network device configuration.[42] In this model, the specific functions of each layer are as follows.

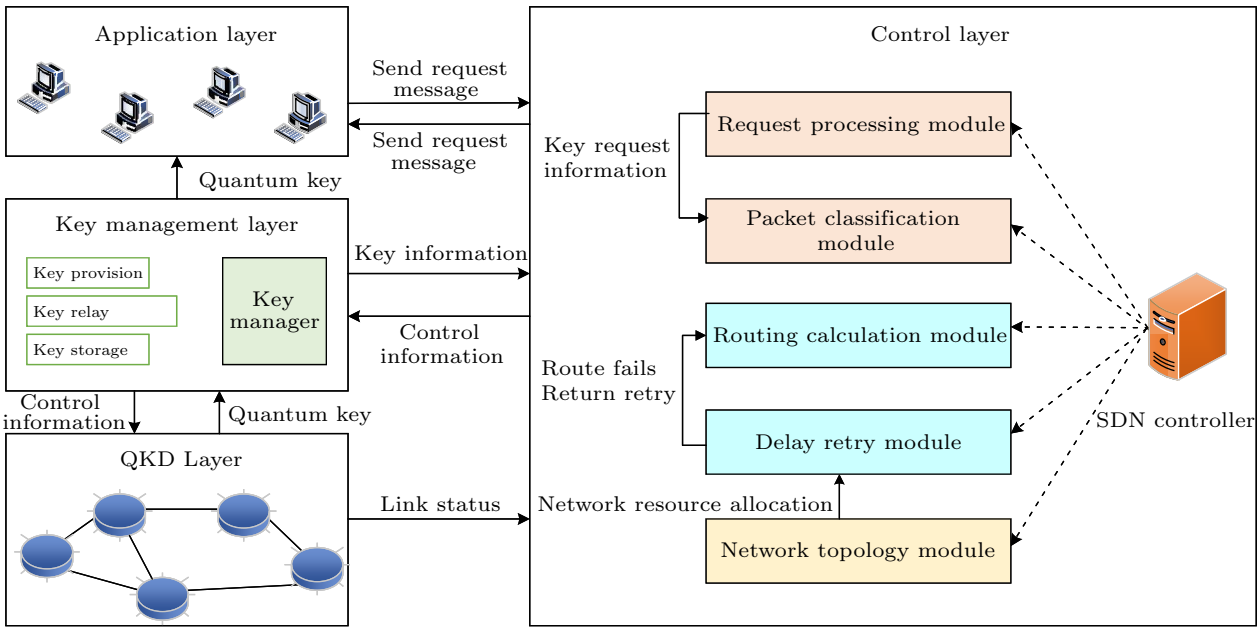


Fig. 3. SDN-based QKD network model.

(i) Application layer. In the application layer, secure key service requests with specific security requirements are generated. The application layer consists of various entities involved in data transmission, serving as the main service provider and user interface of the QKD network. An application program initiates a key request, which is then submitted to the control layer. Upon receiving the key request, the controller enters a blocking and waiting phase. The application program can proceed to the subsequent key generation phase only after receiving the response information from the control layer.

(ii) Control layer. The control layer consists of five modules: request management, packet classification, route calculation, delayed retry, and network topology. Upon receiving a key service request from the application layer, the control layer classifies the packets and forwards them to the route calculation module. If the request routing fails, it is passed to the delayed retry module, which decides whether to rejoin the request queue based on predefined acceptable delay ranges. Requests will be reprocessed without exceeding the delay constraints until successful routing or a timeout occurs. The route calculation module interacts with the network topology module, collecting network resource status and providing a complete network topology structure for path selection.

(iii) Key management layer. The key management layer is responsible for receiving quantum keys generated by the QKD layer and managing the storage of these keys. Additionally, it can facilitate the provision of keys to applications. The key management layer engages in information exchange with the application layer, control layer, and quantum layer.

(iv) QKD layer. This layer is primarily responsible for implementing quantum key distribution and quantum communication functionalities. Its key functions include quantum signal transmission through quantum channels, generation and detection of quantum states, execution of specific protocols for secure key distribution, and the application of error correction codes and privacy amplification to enhance key security and availability. The QKD layer plays a crucial role in ensuring the security and reliability of the keys.

3. Linear programming constraint model and KRC routing algorithm

In this section, a linear programming constrained model for the key resource allocation (KRA) problem in QKD secure optical networks is formulated, and a GKSR scheduling scheme and a micro-log-based key storage algorithm are proposed, enabling effective storage and management of key resources, thereby improving key resource utilization and service efficiency. Finally, a key resource consumption (KRC) routing optimization strategy is proposed to find the optimal relay path for successful key exchange. The set symbols and variable symbols are defined in Tables 1 and 2, respectively.

3.1. A linear programming constraint model for KRA problem

A linear programming constraint model is proposed for addressing the key resource allocation (KRA) problem in quantum key distribution (QKD) secure optical networks. The proposed approach considers two aspects: the constraints of key service requests and preparation for subsequent transmission path optimization schemes. Based on these factors, the linear programming constraint model for QKD secure optical networks is formulated.

Let $G(V,E)$ be a weighted graph used to represent the network, where $V=\{v_i\}$ is the set of nodes in the QKD network, (v_i,v_j) represents a pair of connected nodes, and $E=\{e_{ij}\}$ is the set of links in the QKD network. Each edge $e_{ij}\in E$ is represented as $e_{ij}=(v_i,v_j)$, indicating a QKD link between node pairs (v_i,v_j). The distance between a pair of QKD nodes is defined as $|e_{ij}|=|(v_i,v_j)|$, where $i,j=0,1,\ldots,n-1$, $j\neq i$, $v_i,v_j\in V$. $|V|$ represents the number of nodes, and $|E|$ represents the number of edges in the graph.

Table 1. Set symbol.

Parameter	Description
V	the set of nodes in a QKD network
V_t	the set of relay nodes in a QKD network
E	the set of all QKD links
E_t	the link set with idle channels in time slot t
L	the set of log information records in a QKD network
R	the set of key service requests
R_t	the key service request set being executed in time slot t
P_r	the data packet set of key service request r
$M_{(s_r,d_r)}$	the path set from source node s_r to destination node d_r
T	the set of time slots in a time cycle

Table 2. Variable symbol.

Parameter	Description
s_r	source node for key service request r
d_r	destination node of key service request r
f_r	the amount of keys required for a key service request r
c_r	the actual total amount of keys consumed by a key service request r to complete routing
x_r	boolean variable, if the key request initiated by the user is successful, then $x_r=1$; otherwise, $x_r=0$
Num_r	the number of relay nodes through which a key service request r has been successfully routed
φ_e^t	used for the key generation amount of link $e\in E$ within one time slot t
ϑ_e^t	used for the key consumption amount of link $e\in E$ within one time slot t
g_e^t	the remaining key amount of the quantum key pool in link e under time slot t
$q_{v,e}^t$	represents traffic passing through nodes, where the subscript and superscript respectively indicate the flow passing through a specific node $v\in V$ at a certain time slot $t\in T$, and on a certified key distribution link $e\in E$
h_m	the number of hops in a path m
c_p	the amount of key consumed by a data packet p on the link is classified as 128, 192, and 256 based on the security requirements of the key service request, respectively

Assuming that the source and destination nodes of each data packet in a key service request, as well as the security requirements of the request, are consistent, this paper employs the AES encryption mechanism, where different key lengths represent different security requirements. The key lengths used in the AES encryption algorithm are 128 bits, 192 bits, and 256 bits. Therefore, this paper takes their greatest common divisor, 64 bits, as a unit key length. Additionally, a time slot (t) is defined as the time required to generate a unit key length on the quantum channel of a link. The required key length for a data packet on a link is represented by $n(p)=2$, 3, or 4 unit key lengths, and the corresponding time slot needed is nt.

Objective function for maximizing the number of successful key service requests Based on the aforementioned model, this paper first proposes a model for the maximum number of successful key service requests. For all key service requests within a time period T, the objective function can be represented by

$$\mathrm{Max}\left(\sum_{t\in T}\sum_{r\in R}x_r\right). \tag{1}$$

The objective function constraints of this model are as follows:

$$\frac{\sum_{e\in m}g_e^t}{h_m}\geq 128\,|P_r|\,,\quad \exists m\in M_{(s_r,d_r)},\quad \forall r\in R, \tag{2}$$

$$|m|\leq|E|\,,\quad \forall m\in M_{(s_r,d_r)},\quad \forall r\in R, \tag{3}$$

$$\sum_{e\in m}n(p)\,t\,|r|\leq\sum_{e\in m}|T_{\mathrm{free}}(e)|\,,\quad n(p)=2,3,4,\quad \exists m\in M_{(s_r,d_r)}, \tag{4}$$

$$\sum_{t\in T}\sum_{r\in R}f_r\leq\sum_{t\in T}\sum_{e\in E}g_e^t, \tag{5}$$

$$\sum_{r\in R}|m|\leq|E|\cdot\sum_{r\in R}\mathrm{Num}_r. \tag{6}$$

Equation (2) signifies the necessity for the existence of a path, where the average key quantity on each segment of the QKD link must not be less than the key quantity consumed by the key service request; otherwise, the key service request will be deemed unsuccessful. Furthermore, considering the key scheduling scheme proposed in the next section, 128-bit key length is used as the minimum constraint in this case. g_e^t represents the remaining key quantity in the quantum key pool of link e at time slot t. The remaining key quantity refers to the number of valid keys that can be securely used after key generation, error checking, and error correction steps in the quantum key distribution process on a given link, typically reflected by the amount of unused keys in the virtual key pool.

Equation (3) states that for each key service request r, the number of QKD links it traverses does not exceed the total number of links in the entire topology. Equation (4) represents that for each key service request r, there exists a feasible path such that the total number of time slots required on each link does not exceed the available time slot resources within the current time period T, where $|r|$ denotes the number of data packets in a key service request and $|T_{\mathrm{free}}(e)|$ denotes the number of schedulable time slot resources on the QKD link e within the current time period T.

Equation (5) states that within a time period T, the total amount of key quantity required by the key service request set R should not exceed the total remaining key quantity on all

QKD links e in the entire topology during that time period. Equation (6) states that when deploying a QKD network, the number of QKD links traversed by the key service request set R should not exceed the product of the total number of links in the entire topology and the number of nodes traversed by the key service request set R.

Objective function for minimum key consumption For each approved key service request, this paper further introduces a minimum key consumption model. The objective function for the total amount of key quantity consumed by the key service request set R within a time period T can be represented by

$$\mathrm{Min}\left(\sum_{r\in R} c_r\right),\ \ \forall T. \tag{7}$$

The objective function constraints of this model are as follows:

$$n(p)t \le |T_{\mathrm{free}}(e)|,\ \ n(p)=2,3,4,$$
$$\forall p\in P_r,\ \ \forall e\in m,\ \ \forall m\in M_{(s_r,d_r)}, \tag{8}$$

$$64\cdot n(p)\le g_e^t,\ \ n(p)=2,3,4,$$
$$\forall p\in P_r,\ \ \forall e\in m,\ \ \forall m\in M_{(s_r,d_r)}, \tag{9}$$

$$\sum_{\substack{v_i\in V,\, i\neq j,\\ e\in E,\, t\in T}} q_{v,e}^t \ - \sum_{\substack{v_j\in V,\, i\neq j,\\ e\in E,\, t\in T}} q_{v,e}^t = \begin{cases} \sum\limits_{\substack{v_j\in V,\, i\neq j,\\ e\in E,\, t\in T}} q_{v,e}^t, & i=s_r,\\ -\sum\limits_{\substack{v_j\in V,\, i\neq j,\\ e\in E,\, t\in T}} q_{v,e}^t, & i=d_r,\\ 0, & \text{otherwise}, \end{cases} \tag{10}$$

$$Q_{(t\in T,r\in R)} = \rho\cdot \mathrm{QBER} - \sigma\cdot\frac{\mathrm{Qcc}}{1000},\ \ (\rho>\sigma). \tag{11}$$

Equation (8) states that the number of time slots required by approved key service requests on each QKD link e should not exceed the available slot resources on that link within the current time period T. Equation (9) indicates that the amount of key quantity required by approved key service requests on each QKD link should not exceed the remaining key quantity on that link under t time slots. Equation (10) represents the traffic conservation equation, denoting the QKD path flow constraints for all nodes $t\in T$. Here, the symbol $(+)$ signifies inflow, while $(-)$ denotes outflow. With the exception of the source and destination nodes, it is mandated that the outflow of traffic from each node equals its inflow.

Equation (11) represents the quality of quantum channel Q, which is mainly related to the quantum bit error rate (QBER) and quantum channel capacity (Qcc). The two weighting factors ρ and σ are used to balance the importance of these two factors. As QBER has a greater impact on the overall channel quality, it is ensured that $\rho>\sigma$. The occurrence of errors in the quantum channel is attributed to two factors: system imperfections and dark counts. According to the system security theory of the BB84 protocol, the quantum bit error rate (QBER) can be calculated using the formula $\mathrm{QBER}=\frac{e\cdot(N\cdot\eta\cdot\omega_{\mathrm{net}})+d}{N\cdot\eta\cdot\omega_{\mathrm{net}}}$, where ω_{net} represents the transmission loss as defined in Eq. (15) below, d denotes the dark count rate of detectors (set at 100), N represents the number of photons transmitted per second following a Poisson distribution with a parameter $\lambda=5\times10^5$, and η represents the probability of successful detection for each photon, set at 10%. The quantum channel capacity (Qcc) is related to the transition probability $P(b|a)$ of the channel. When the channel quality exceeds a certain threshold (taken as 0.01), it indicates that the link no longer meets the conditions for information transmission and should be excluded during route selection

$$U=\frac{\sum_{t\in T}(\varphi_e^t-\vartheta_e^t)}{T},\ \ e\in E, \tag{12}$$

$$\left|\sum_{\alpha T} U\right| \le \Omega,\ \ \forall e\in m,\ \ \forall m\in M_{(s_r,d_r)}. \tag{13}$$

In order to maintain the stability of the number of keys in the key pool, this model proposes the concept of key pool update rate and defines a threshold Ω. If the sum of the key pool update rates for a link exceeds the threshold, the link is not considered for forwarding encrypted data packets. Equation (12) represents the key pool update rate, which indicates the variation of the number of keys in the key pool on a specific link within a time period T. If $U>0$, it indicates that the number of keys in the key pool is increasing during the time period T, whereas if $U<0$, it indicates a decrease in the number of keys. Equation (13) represents the constraint logic for Eq. (12), where α denotes the coefficient corresponding to the time period T elapsed since the initiation of the network.

3.2. Key storage and resource allocation scheme

3.2.1. Key resource scheduling scheme based on key security requirement classification

In QKD networks, the key generation rate is generally low, and key resources are scarce. By constructing a virtual quantum key pool, key resources can be stored in a short period of time to promptly respond to key service requests, reduce end-to-end key negotiation latency, and achieve effective adaptation with network operations. The quantum key pool can monitor real-time key information and handle symmetric keys in pairs. This paper assumes that all relay nodes are trusted relay nodes, therefore, the process of accessing keys through the virtualized quantum key pool should be secure for data packets. To further improve key access efficiency and key resource utilization, this paper proposes a key resource

scheduling (GKSR) scheme based on key security requirement classification for the virtual quantum key pool.

Currently, the widely used encryption mechanism in QKD networks is the one-time-pad (OTP). OTP provides a key of the same length as the data packet for each encryption process, and this key is only used once, achieving perfect secrecy. However, OTP still consumes a considerable amount of key resources. A more practical solution is to utilize the advanced encryption standard (AES) algorithm. The AES algorithm can encrypt and decrypt data packets using keys of 128 bits, 192 bits, or 256 bits. This paper proposes a spatially partitioned quantum key pool model, as shown in Fig. 4, which divides the quantum key pool into three key spaces. The generated keys are randomly stored in these three key spaces, and if the keys in a certain key space reach the MAX threshold, they are injected into other key spaces until the quantum key pool is full.

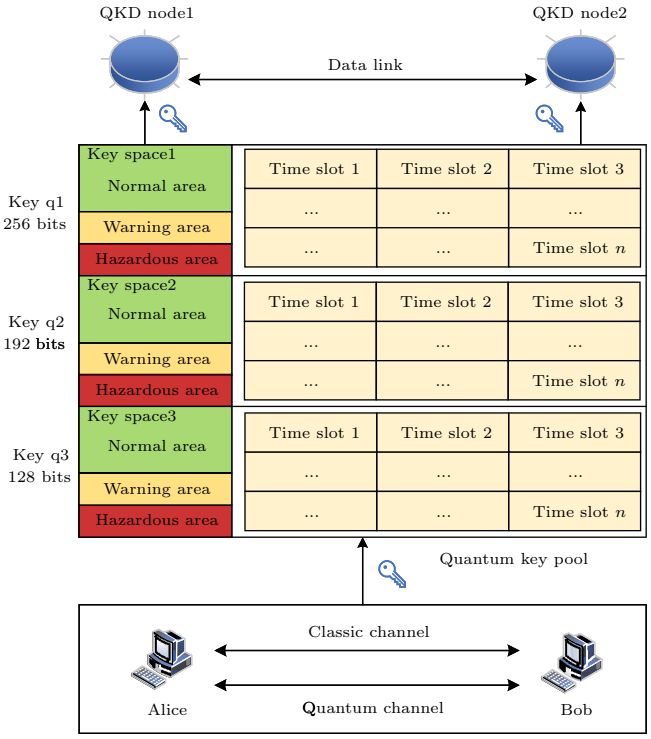


Fig. 4. Quantum key pool model for delimited space.

The key pool can provide quantum keys for key service requests of different security levels. Key space 1 provides 256-bit keys for the highest security level key service requests. Key space 2 provides 192-bit keys for the second-highest security level key service requests. Key space 3 provides 128-bit keys for the lowest security level key service requests.

When the remaining key quantity in the corresponding key space for a key service request is insufficient, it will not be possible to guarantee the normal transmission of data packets. Therefore, this paper proposes the concept of secure degradation of key service requests. When the remaining key quantity in higher priority key spaces is insufficient, the key quantity from the next key space is used to reduce the restriction of quantum key resources on data packet transmission and improve key resource utilization. The consumption of quantum keys during the transmission of data packets with different security levels is shown in Table 3.

Table 3. Quantum key consumption table for packets with different security levels.

	Key q1	Key q2	Key q3	Is it safe to downgrade	Send a warning message
1	false	false	false	×	key pool exhausted
2	false	false	true	√	do not receive maximum security packets
3	false	true	false	√	do not receive minimum security packets
4	false	true	true	√	
5	true	false	false	×	do not receive packets with the next highest or lowest security level
6	true	false	true	√	
7	true	true	false	×	do not receive minimum security packets
8	true	true	true	×	

When a key is present in a key space, it is recorded as true; when the key is exhausted, it is recorded as false. When the remaining key quantity in all three key spaces is zero, a warning message that indicating the key pool is exhausted needs to be sent to the controller, as in group 1. In groups 2, 3, 4, and 6, when the key quantity in a higher priority key space is insufficient to ensure the normal transmission of data packets, secure degradation is performed to reduce key consumption and improve key resource utilization. However, to ensure the security and reliability of data transmission, secure degradation can only reduce the security level by one, as in group 2. If the key quantity in key space 1 is insufficient, the highest security level data packets are no longer accepted, and the next lower security level data packets are securely degraded and use the keys in key space 3. The specific process for key resource scheduling is shown in Fig. 5.

Each key space is divided into multiple time slots using optical time-division multiplexing (OTDM) technology, and each time slot can generate a key of unit length (64 bits). Consequently, $n(p)$ time slots (where $n(p) = 2, 3, 4$) can be allocated to provide keys for data packets in the key service request. Additionally, this paper refers to the order in which quantum keys are stored in the key space as key freshness. Keys with lower key freshness will be given priority for use, meaning that keys will be allocated to key service requests based on a queuing strategy.

3.2.2. Quantum key resource allocation scheduling scheme

In this section, a micro-log based key storage algorithm (MLKSA) is proposed as shown in Fig. 6. The usage of quantum keys is recorded using log technology, which includes

key length, security level, and usage time. The log follows a first-in–first-out (FIFO) queue strategy. Initially, it is assumed that the quantum key pools of all links are full. The key pool queries the log queue for the length of the next required key. At this point, a log record is popped from the queue. The quantum key pool reads the log information and generates a key of the corresponding length, ensuring asynchronous ordering and continuity between key generation and usage.

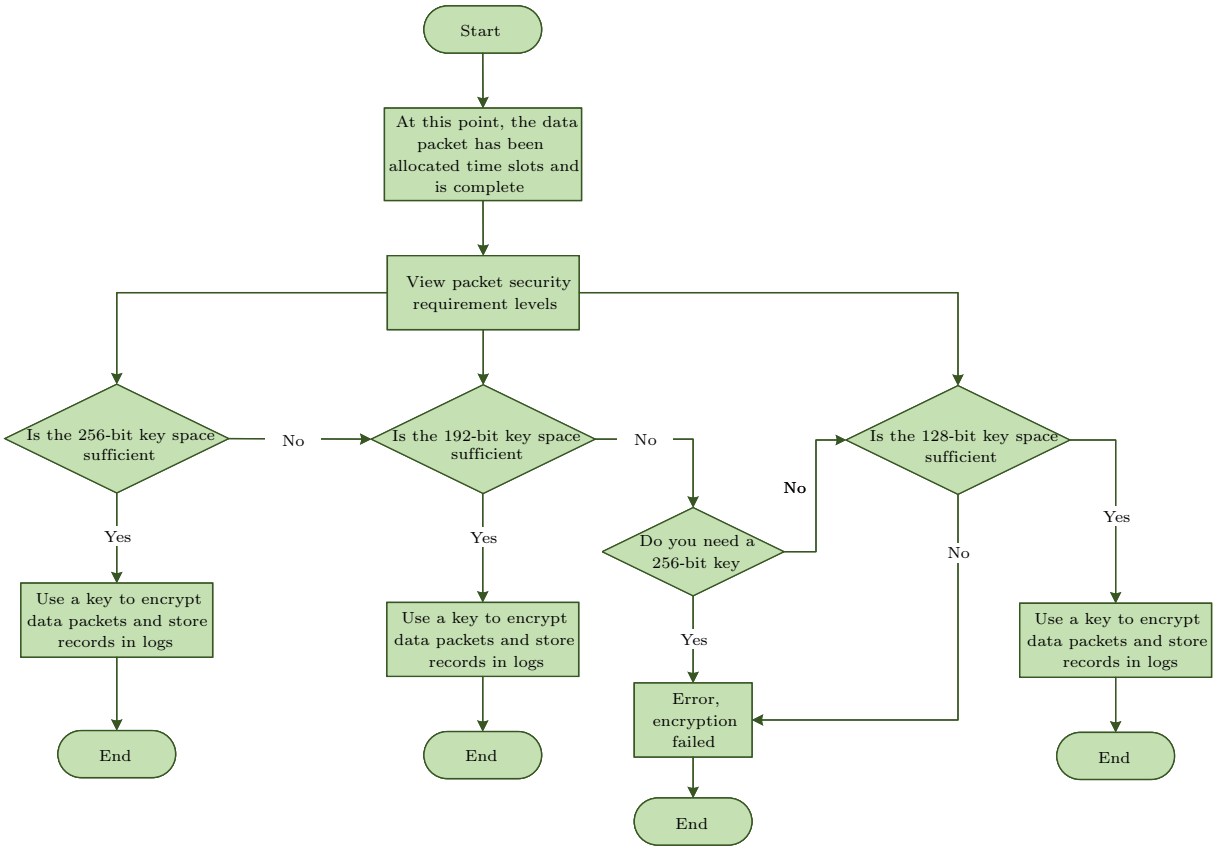


Fig. 5. Key resource scheduling scheme flowchart.

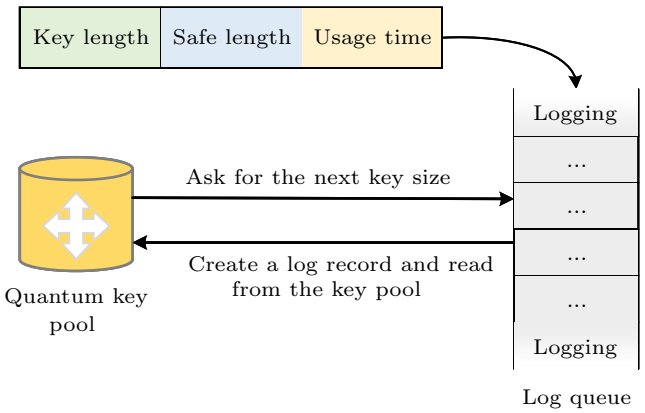


Fig. 6. Micro-log based keystore procedure.

Table 3 illustrates the detailed process of the micro-log based key storage algorithm. Firstly, in lines 5 and 6, the remaining key quantity g_e^t of the quantum key pool for link e is obtained for the time slot t, and the corresponding log queue KL of the quantum key pool is queried. In lines 7–18, if the queue is empty, the loop is exited. If the queue is not empty, the log records are traversed. Based on the retrieved information from the log records, including the length of the used key, security level, and usage time, the corresponding quantum key of that length is generated using a key distribution protocol. For example, if a key length of 128 bits and security level of 1 are obtained, the quantum key pool generates a 128-bit quantum key for storage. This process continues by iterating through the log records until the quantum key pool becomes full. At this point, the thread sleeps, and it is awakened when the keys in the quantum key pool are utilized again.

3.3. KRC routing optimization algorithm

The primary objective of the QKD network routing algorithm is to improve the key utilization rate during the process of quantum key distribution and reduce the consumption of quantum key resources. In light of this objective, this paper proposes a key resource consumption (KRC) routing optimization algorithm to find the optimal path from one node to another within the network. The routing scheme proposed in this paper allows the QKD network to select suitable key relay paths based on the conditions of the QKD system, thereby reducing additional key consumption and optimizing the utilization of key resources in the QKD network.

Algorithm 1: key storage algorithm based on micro-log technology.

Input: E, V, Vt, T, g_e^t, L, n used QKD modules and quantum channels
Output: Resource assignment for storing keys

```
Sort V with the length of shortest path in ascending order
for each node pair v ∈ V do
  for each time-slot nt ∈ T do
    for index l ∈ L of each possible key rate do
      Get the remaining key quantity g_e^t of QKP
      Get the key usage log KL of QKP
      if KL = ∅ break
      wake up
      Traverse KL
        if the used key length == 128
        then g_e^t += 128
        if the used key length == 192
        then g_e^t += 192
        if the used key length == 256
        then g_e^t += 256
        if g_e^t == the capacity of QKP
          sleep
      end
    end
  end
end
```

If the channel quality during the key generation process is sufficiently good, the rate of quantum key generation will also improve, and the rate of quantum key generation is the most important parameter for measuring the performance of a quantum link. However, in the case of long-distance quantum signal transmission, the rate of quantum key distribution is affected by transmission losses, resulting in a significant decrease in the key generation rate. This paper primarily considers the impact of transmission distance on the rate of quantum key generation. Equation (14) represents the average rate of quantum key generation as follows:

$$R_{\text{sift}} = N\eta\omega - d, \tag{14}$$

$$\omega_{\text{net}} = 10^{-\frac{l_f|e_{ij}|}{10}}, \tag{15}$$

where N is the number of photons emitted by the transmitter per second, and represents the efficiency of a single photon detector, set at 10%. Equation (15) represents the transmission loss, denoted as ω_{net}, which is the fiber transmission loss factor. At a wavelength of 1550 nm, the loss of a single-mode fiber is typically around 0.25 dB/km. As the transmission distance increases, it introduces more transmission losses, resulting in a gradual decrease in the rate of quantum key generation.

In addition, this paper stores the quantum keys in three different key spaces with varying security requirements, namely q1, q2, and q3. The required key lengths for q1, q2, and q3 are 256 bits, 192 bits, and 128 bits, respectively, based on the security level desired for the data packets. Next, the maximum common factor of 64 bits is taken as a unit key length. The time slot, denoted as t, is defined as the time required to generate a unit key length on the quantum channel of the link. The key length required for a data packet on the link is $(n(p) = 2,\ 3,\ 4)$ unit key lengths. Finally, the number of 64-bit key blocks obtained from different key spaces is determined by (denoted as a, b, c)

$$g_e^t = (a+b+c)\cdot 64 \geq \text{Threshold},\quad t \in T, \tag{16}$$

$$\tau_T = \sum_{t\in T} \varphi_e^t = \sum_{t\in T} R_{\text{sift}}(t), \tag{17}$$

$$C_e = g_e^t \cdot \ln \tau_T \cdot U, \tag{18}$$

$$y = \sum_{t\in T}\sum_{e\in m} C_e,\quad \forall m \in M_{(s_r,d_r)}. \tag{19}$$

Within a time period, the remaining key quantity g_e^t can be calculated as shown in Eq. (16), where the Threshold represents the threshold of the remaining key quantity for each key space in each QKD link. This threshold is set by the highest authority of the entire network. Equation (17) represents the generation status of quantum keys for link e, denoted as τ_T. The $\sum_{t\in T} R_{\text{sift}}(t)$ represents the quantity of keys generated within a time period. Equation (18) defines the weight of each link as C_e. In Eq. (19), the variable y represents the calculation method for the weight of each path from the source node to the destination node, taking into account factors such as the generation status of quantum keys within a time period T, key pool update rate, and the remaining key quantity in each key space

$$\text{Min}\left(y + \varepsilon y_\theta + \varepsilon^2 y_{\theta''} + \varepsilon^3 y_{\theta'''} + \cdots + \varepsilon^{k-1} y_{\theta^{k-1}}\right). \tag{20}$$

During the routing process, when each intermediate node forwards a data packet, the influence of the link with a greater number of hops to the destination node on the routing selection is typically diminishing. Therefore, a decrement factor ε $(0 < \varepsilon < 1)$ is defined in this paper. The variable E_v^{k-1} refers to the subset of links for node v at the k-th hop. The variable θ^{k-1} refers to a specific node in the subset of nodes at the k-th hop from the source node. In this paper, the routing selection will be calculated based on the path weight according to Eq. (20)

$$m' = \begin{cases} \exists! e \notin m,\ h_m = h_{m'}, \\ h_{m'} = h_m + 1, \\ h_{m'} = h_m - 1, \end{cases} \quad m' \in M'_{(s_r,d_r)}. \tag{21}$$

Equation (21) represents the element rule of the neighborhood path set M' for path m, where h_m denotes the number of hops for path m, m' denotes an element in the neighborhood path set, and e refers to a specific link in path m

$$\left|m^\Delta\right| > \min\left(\left|M'_{(s_r,d_r)}\right|\right), \tag{22}$$

$$\gamma \leq \max\left(\left|M'_{(s_r,d_r)}\right|\right). \tag{23}$$

When continuously iterating to find the optimal path, the termination conditions are determined based on Eqs. (22) and (23). The first condition is that the selected best path m^Δ has

the minimum number of hops in the network topology, considering the source and destination nodes. The second condition is that the iteration count γ should not exceed the maximum hop count of paths in the topology when considering the source and destination nodes.

Figure 7 presents the overall resource allocation scheme and routing optimization algorithm. Firstly, after the completion of filtering, R_t and E_t are used to compute the set of all paths from the source node s_r to the destination node d_r for $r \in R_t$, denoted as $M_{(s_r,d_r)}$. One path m is randomly selected from $M_{(s_r,d_r)}$. Next, a search is conducted to find all paths that have minimal differences compared to path m. These differences include paths with one additional hop, paths with one fewer hop, and paths with variations in the sequence of links within the path. These paths collectively form the set $M'_{(s_r,d_r)}$. The best path m^{Δ} is calculated from the set $M'_{(s_r,d_r)}$ using Eq. (20), where $\Delta = 1$. The iteration continues until the following conditions are met, leading to the termination of the loop. 1) The selected best path has the minimum number of hops. 2) The iteration count exceeds the maximum hop count value. Through this process, the optimal path m^{Δ} is computed.

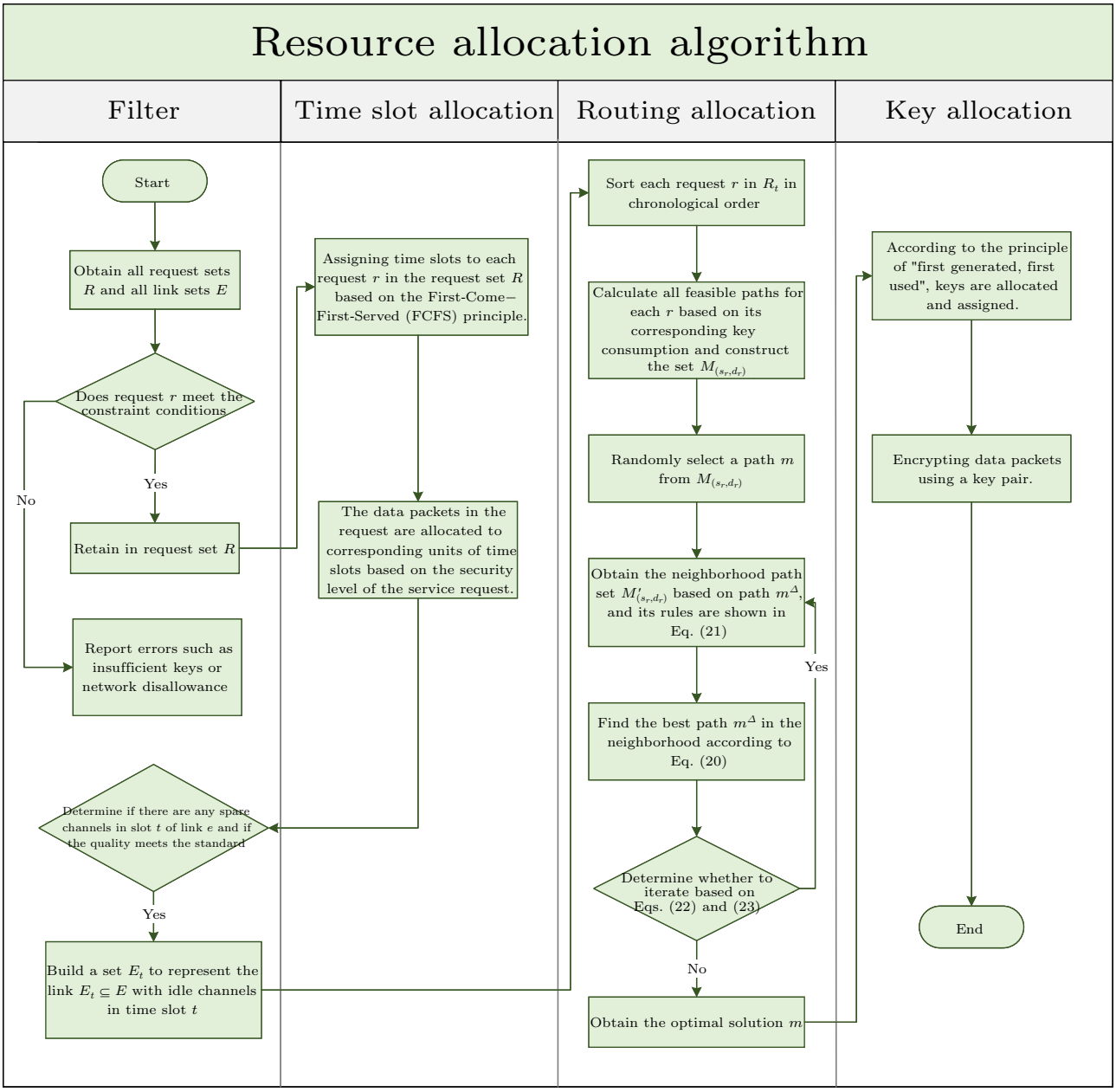


Fig. 7. Overall resource allocation plan and routing optimization algorithm flowchart.

4. Simulation experiments and analysis

In this section, extensive simulations were conducted to verify the KRC routing optimization algorithm proposed in this paper using IntelliJ IDEA. In the simulation experiment, two network topologies, NSFNET (14 nodes and 22 links) and USNET (24 nodes and 43 links), were used. A time period T

of 500 ms was set in this study, and the number of data packets in each request was randomly varied between 1 and 30. The source and destination nodes of each data packet in a key service request were set to be the same. With a fixed number of key service requests, the source and destination nodes for forwarding requests were randomly selected. The remaining threshold is set as Threshold = 640, $\rho = 0.8$, $\sigma = 0.2$, $\Omega = 5000$, $\varepsilon = 0.8$.

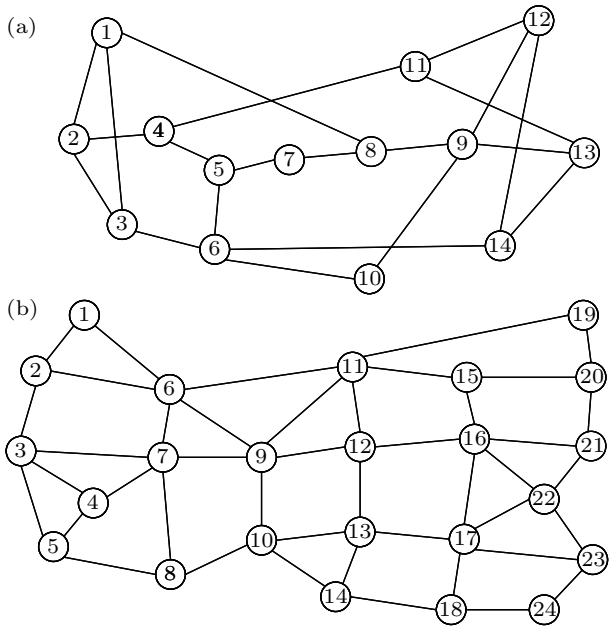


Fig. 8. Network topology used for simulation (a) NSFNET and (b) USNET.

As shown in Fig. 8, the simulation experiment compares the residual key quantity priority routing algorithm (RKQ),[43] shortest path routing algorithm (KSP),[44] and random routing algorithm (RRA).[45] The evaluation includes the security performance, packet loss rate, key distribution success rate, time delay, and key resource utilization

$$\mathrm{SR} = 1 - \frac{\mathrm{SPDP}}{\mathrm{TSSP}}, \tag{24}$$

$$\mathrm{PLR} = \left(1 - \frac{\mathrm{TSTP}}{\mathrm{TP}}\right) \times 100\%, \tag{25}$$

$$\mathrm{KSR} = \frac{\mathrm{TPSRD}}{\mathrm{TPRFA}} \times 100\%, \tag{26}$$

$$\mathrm{RTT} = \frac{\sum_{r\in R}\sum_{p\in r}\mathrm{rtt}}{\mathrm{TSTP}}, \tag{27}$$

$$\mathrm{KRUR} = \left(1 - \frac{\mathrm{TKCDP}}{\mathrm{TKCP}}\right) \times 100\%. \tag{28}$$

The calculation formulas for security performance, packet loss rate, key distribution success rate, time delay, and key resource utilization are given by Eqs. (24)–(28) respectively. This paper adopts a scoring mechanism to evaluate the overall security performance of the network. A score is assigned to each data packet, and the security performance is reflected based on the score rate. The packet loss rate is calculated as the total number of discarded packets divided by the total number of forwarded packets. To ensure the physical layer security of the network, QKD technology is embedded in our proposed solution. However, data packets with security requirements are subjected to the constraint of quantum key quantity during transmission. In traditional software-defined networks, data packets are dropped in the event of link failures, routing errors, congestion, and other similar situations. After incorporating QKD technology, additional factors leading to network congestion have been introduced. For instance, improper routing policies can cause data packets to wait at intermediate nodes due to the occupation of quantum channels. This congestion is particularly likely to occur when multiple data packets need to be forwarded through a single link, resulting in packet loss. In traditional software-defined networks, under the scenario of initiating hundreds of requests per second, the packet loss rate is almost 0. After incorporating QKD technology, the limited rate of link key generation significantly increases the likelihood of congestion at intermediate nodes, resulting in a higher packet loss rate. This also affects the overall network lifespan and service quality.[46]

Table 5. Symbols and definitions of experimental variables.

Symbol	Description
SP	safety performance
PLR	packet loss rate
KSR	key distribution success rate
RTT	time delay
KRUR	key resource utilization rate
TSSP	total score of safety performance
SPDP	safety performance deduction points
TP	total number of packets
TSTP	total number of successfully transmitted packets
TPSRD	the total number of packets successfully reaching the destination node
TPRFA	the total number of packets required to be forwarded for all approved key service requests
TKCP	the total number of keys consumed by the data packet
TKCDP	the total number of keys consumed by discarded packets
rtt	the round-trip latency of a data packet transmission

The key distribution success rate refers to the ratio of successful routing and key distribution for the set of key service requests R, achieving secure key sharing between source and destination nodes. In the simulation, a successful route finding and forwarding is considered as a successful key distribution, without considering the cases where the distribution may fail due to eavesdroppers or other physical factors. The time delay represents the average transmission delay of all successfully forwarded packets during the experimental process.

The key resource utilization rate is the ratio of the total amount of key consumed by successfully forwarded packets to the total amount of key consumed by all packets. When the key resources in a QKD network are used efficiently, the key resource utilization rate $(0 \leq \mathrm{KRUR} \leq 1)$ reaches its maximum value. The smaller the packet loss rate and time delay,

the higher the quality of service during the experimental process. A higher request success rate and key resource utilization rate indicate that more requests were approved while the key resources were used as efficiently as possible.

This paper considers the flexible adjustment of the key length for the AES algorithm to alleviate the issues of low quantum key generation rate and insufficient key in QKD networks. In this paper, the degradation count is scored to evaluate the overall security performance of the network. It is assumed that each degradable packet can be downgraded three times during end-to-end transmission in the path. Downgrading from a key length of 256 bits to 192 bits results in a deduction of 10 points, while downgrading from a key length of 192 bits to 128 bits results in a deduction of 30 points. The lower the score, the less secure it is. The security performance in the NSFNET topology (a) and the USNET topology (b) is depicted in Fig. 9.

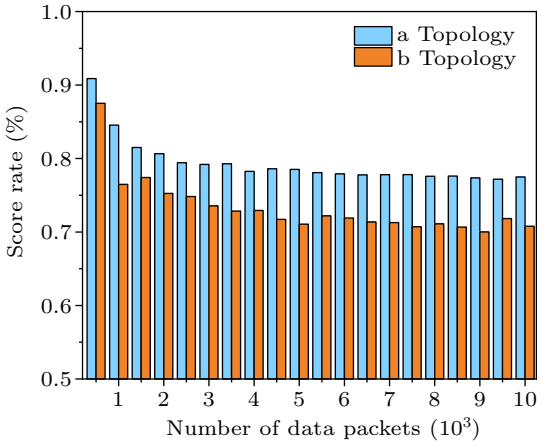


Fig. 9. Network security performance as a function of packet volume: (a) NSFNET, (b) USNET.

According to the data analysis, it can be observed that when the packet volume exceeds approximately 2500, an average of one degradation occurs during packet transmission along the path. In other words, in each path transmission of a data packet, it is expected to traverse a link with a corresponding insufficient key length. As the volume of packet transmission gradually increases in the network topology, the degradation count score fluctuates between 70% and 80%, indicating relatively good performance.

Assuming a fixed initial quantity of 100 key units in each quantum key pool, the relationship between the packet loss rate and the number of QKD key service requests for the four routing algorithms under different network topologies is depicted in Fig. 10. In the NSFNET topology, compared to the RKQ, KSP, and RRA routing algorithms, the packet loss rate has been reduced by 9.08%, 11.05%, and 15.2%, respectively. It can be observed that as the number of QKD key service requests increases, the packet loss rate also increases. This is due to congestion caused by the large number of packets. However, the KRC routing optimization algorithm performs relatively well in terms of packet loss.

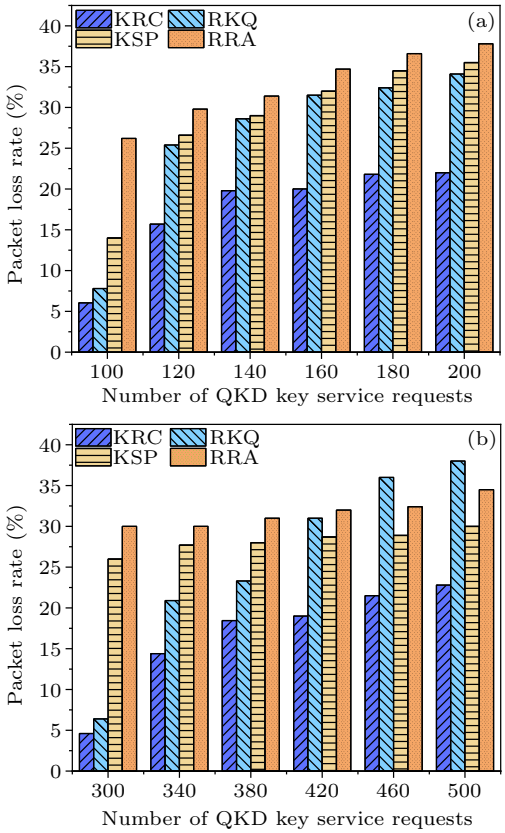


Fig. 10. Packet loss rate as a function of the number of service requests for different QKD keys: (a) NSFNET, (b) USNET.

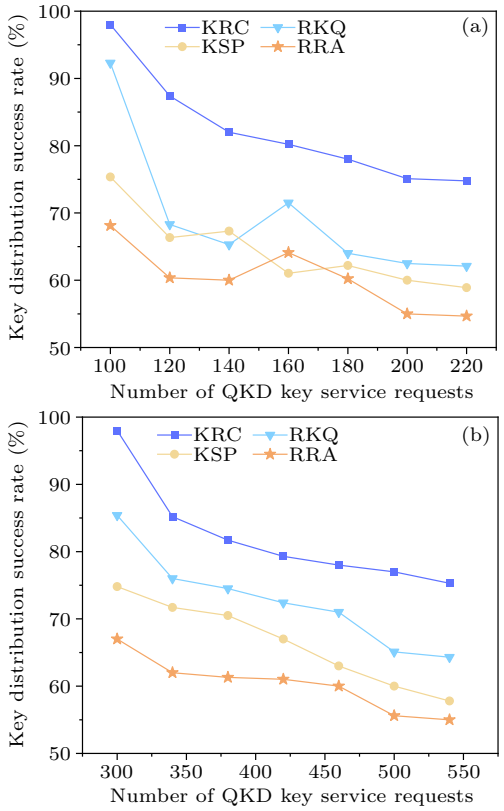


Fig. 11. Key distribution success rate as a function of the number of key service requests for different QKDs: (a) NSFNET, (b) USNET.

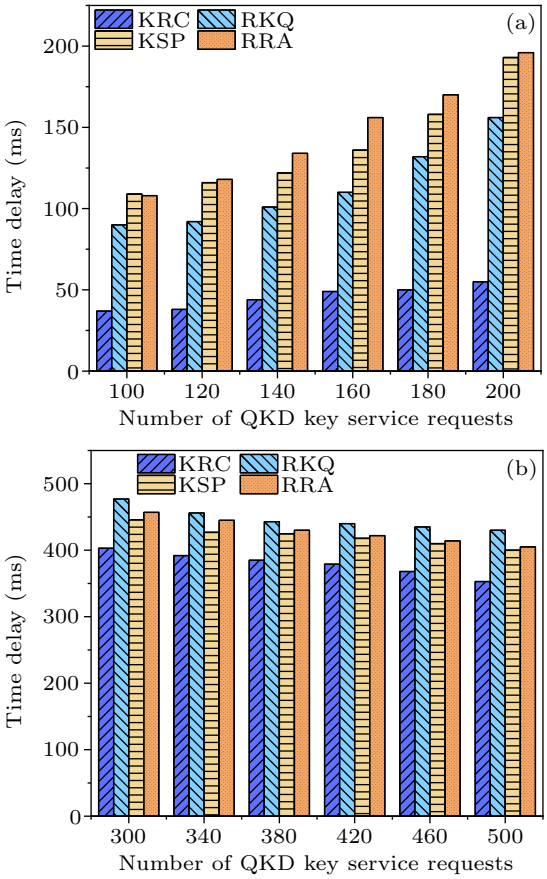


Fig. 12. Time delay as a function of the number of service requests for different QKD keys: (a) NSFNET, (b) USNET.

Furthermore, the experiment also compares the relationship between the key distribution success rate of each routing algorithm and the number of QKD key service requests under two network topologies, as shown in Fig. 11. It can be observed that the KRC routing optimization algorithm achieves the highest key distribution success rate, reaching 94.5%. Compared to the RKQ, KSP, and RRA routing algorithms in the NSFNET topology, the key distribution success rate has improved by 10.3%, 18.65%, and 22.64%, respectively. A higher key distribution success rate implies a higher frequency of successfully establishing shared keys between source and destination nodes during the process of quantum key distribution, indicating greater reliability and stability. However, as the number of key service requests increases, the key distribution success rate also decreases.

In addition, the time delay of each routing algorithm is also compared, as shown in Fig. 12. In the USNET network topology, compared to the RKQ, KSP, and RRA routing algorithms, the time delay has been reduced by 8.09%, 2.45%, and 4.23%, respectively. It can be observed that the time delay is lower in the NSFNET network topology, while the USNET network exhibits higher time delay. This is due to the increase in the number of QKD key service requests, where a portion of the packets do not receive timely responses and remain in a state of delayed retries for an extended period, resulting in increased time delay.

Finally, a comparison of the key resource utilization rate for each routing algorithm is conducted under the two network topologies, as shown in Fig. 13. It can be observed that as the number of QKD key service requests increases, the utilization rate of key resources also increases. Compared to the RKQ, KSP, and RRA routing algorithms in the USNET network topology, the key resource utilization rate has improved by 11%, 23.31%, and 8.74%, respectively. It is evident that the KRC routing optimization algorithm has a high key resource utilization rate and demonstrates excellent performance.

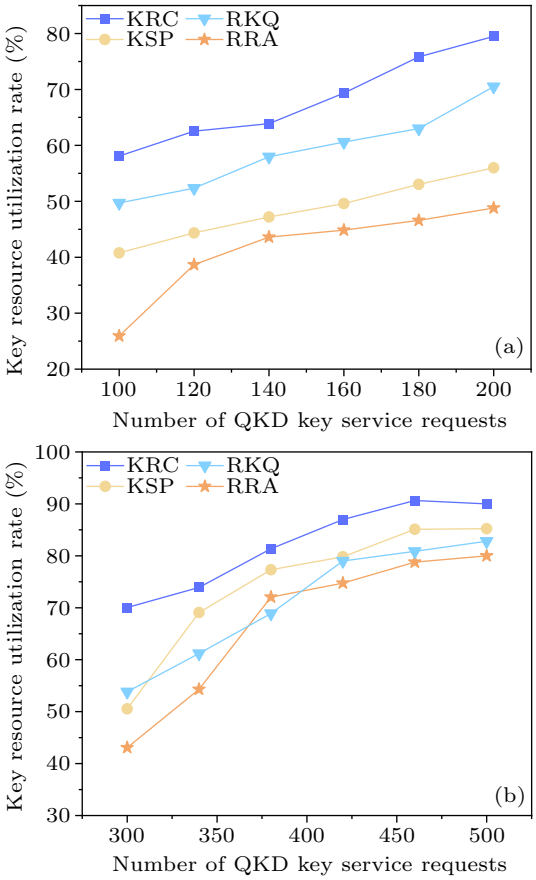


Fig. 13. Key resource utilization rate as a function of the number of key service requests for different QKDs: (a) NSFNET, (b) USNET.

This article has made improvements to the key storage algorithm and proposed the micro-log-based key storage algorithm (MLKSA). Simulation experiments are conducted using 100, 300, and 500 data packets, and compare with the traditional key storage algorithm (TKSA). The simulation results are shown in Fig. 14. As time increases, the remaining key quantity in the quantum key pool gradually decreases, showing different performance for different numbers of data packets. It can be observed that with an increase in the number of data packets, the key consumption rate also increases, resulting in a reduced remaining key quantity. The MLKSA has shown an improvement in the remaining key quantity of 8.3% to 12.6% compared to TKSA.

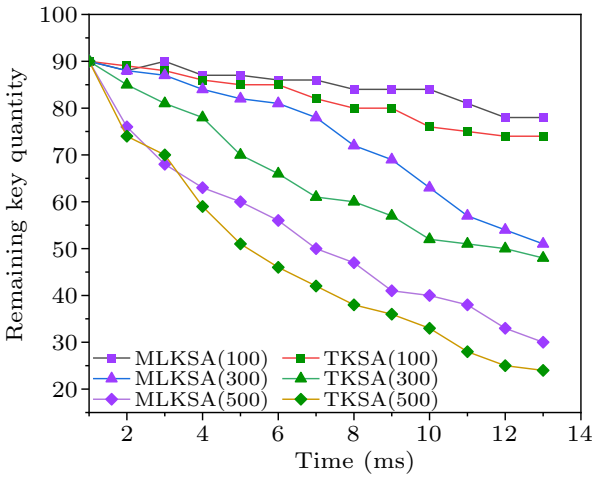


Fig. 14. Key consumption for different packets.

5. Conclusion

With the rapid development of quantum technology, there is an urgent need to design efficient quantum key distribution (QKD) network schemes for secure key sharing. In this work, we focus on studying a novel QKD resource allocation and routing scheduling model and algorithm. To address the key resource allocation (KRA) problem, we first formulate a linear programming constraint model, taking into account the constraints of key service requests and preparing for the optimization of constructing transmission paths. Additionally, we propose the GKSR scheduling scheme and the micro-log-based key storage algorithm to effectively store and manage key resources. Finally, the proposed KRC routing optimization algorithm selects routes by calculating path weights based on the dynamically changing network topology. A comparative analysis is conducted with the RKQ algorithm, the KSP algorithm, and the RRA algorithm. The simulation results indicate that the KRC routing optimization scheme exhibits certain performance improvements compared to the other three routing schemes. Therefore, the new QKD resource allocation and routing scheme proposed in this paper performs well in scenarios where there are key service requests with multiple security requirements and scarce key resources. This is of significant importance for enhancing the quality of QKD network services. Quantum key distribution is an important research field, and the efficiency of quantum key distribution in dynamic scenarios and network security issues will be future research directions. In future work, particular attention can be given to studying topics such as efficient utilization of quantum key resources and key protection strategies.

Acknowledgment

Project supported by the Natural Science Foundation of Jilin Province of China (Grant No. 20210101417JC).

References

[1] Bonnetain X, Naya-Plasencia M and Schrottenloher A 2019 *IACR Trans. Symmetric Cryptol.* **2019** 55
[2] Monz T, Nigg D, Martinez E A, Brandl M F, Schindler P, Rines R, Wang S X, Chuang I L and Blatt R 2016 *Science* **351** 1068
[3] Zhang Q, Xu F, Chen Y A, Peng C Z and Pan J W 2018 *Opt. Express* **26** 24260
[4] Wang H, Zhao Y, Li Y, Yu X, Zhang J, Liu C and Shao Q 2018 *Opt. Fiber Technol.* **45** 195
[5] Boaron A, Boso G, Rusca D, Vulliez C, Autebert C, Caloz M, Perrenoud M, Gras G, Bussières F, Li M J, Nolan D, Martin A and Zbinden H 2018 *Phys. Rev. Lett.* **121** 190502
[6] Bennett C H and Brassard G 2014 *Theor. Comput. Sci.* **560** 7
[7] Dynes J F, Tam W W S, Plews A, Fröhlich B, Sharpe A W, Lucamarini M, Yuan Z, Radig C, Straw A, Edwards T and Shields A J 2016 *Sci. Rep.* **6** 35149
[8] Wang C, Kon W Y, Ng H J and Lim C C 2022 *Light Sci. Appl.* **11** 268
[9] Luo W, Cao L, Shi Y, Wan L, Zhang H, Li S, Chen G, Li Y, Li S, Wang Y, Sun S, Karim M F, Cai H, Kwek L C and Liu A Q 2023 *Light Sci. Appl.* **12** 175
[10] Wang S 2022 *Light Sci. Appl.* **11** 301
[11] Diamanti E, Lo H K, Qi B and Yuan Z 2016 *NPJ Quantum Inf.* **2** 16025
[12] Cao Y, Zhao Y, Wu Y, Yu X and Zhang J 2018 *J. Light. Technol.* **36** 3382
[13] Cao Y, Zhao Y, Wang J, Yu X, Ma Z and Zhang J 2019 *IEEE Commun. Mag.* **57** 152
[14] Cao Y, Zhao Y, Wang J, Yu X, Ma Z and Zhang J 2019 *Opt. Express* **27** 6892
[15] Mehic M, Niemiec M, Rass S, Ma J, Peev M, Aguado A, Martin V, Schauer S, Poppe A and Pacher C 2020 *ACM Comput. Surv. (CSUR)* **53** 1
[16] Cao Y, Zhao Y, Yu X and Wu Y 2017 *J. Opt. Commun. Netw.* **9** 995
[17] Zhao Y, Cao Y, Wang W, Wang H, Yu X, Zhang J, Tornatore M, Wu Y and Mukherjee B 2018 *IEEE Commun. Mag.* **56** 130
[18] Cao Y, Zhao Y, Colman M C, Yu X and Zhang J 2017 *Opt. Express* **25** 26453
[19] Tsai C W, Yang C W, Lin J, Chang Y C and Chang R S 2021 *Appl. Sci.* **11** 3767
[20] Ding H J, Liu J Y, Zhang C M and Wang Q 2020 *Quantum Inf. Process.* **19** 60
[21] Liu J Y, Ding H J, Zhang C M, Xie S P and Wang Q 2019 *Phys. Rev. Appl.* **12** 014059
[22] Ma W, Liu L, Chen B, Gao M, Chen H and Wu J 2020 2020 *Asia Communications and Photonics Conference (ACP) and International Conference on Information Photonics and Optical Communications (IPOC)*, October 24–27, 2020, Beijing, China, p. 1
[23] Chen L Q, Zhao M N, Yu K L, Tu T Y, Zhao Y L and Wang Y C 2021 *Quantum Inf. Process.* **20** 309
[24] Yu X, Liu Y, Zou X, Cao Y, Zhao Y, Nag A and Zhang J 2022 *J. Light. Technol.* **40** 3530
[25] Chen L, Zhang Z, Zhao M, Yu K and Liu S 2022 *Entropy* **24** 1519
[26] Chen L, Chen Q, Zhao M, Chen J, Liu S and Zhao Y 2022 *Entropy* **24** 149
[27] Sharma P, Bhatia V and Prakash S 2022 *Opt. Fiber Technol.* **68** 102755
[28] Zhang Q, Liu Y, Yu X, Zhao Y and Zhang J 2022 *Photonics* **9** 239
[29] Sharma P, Bhatia V and Prakash S 2020 2020 *IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, December 14–17, 2020, New Delhi, India, p. 1
[30] Zhang K, Yu X, Wang Y, Li Y, Zhao Y and Zhang J 2021 2021 *19th International Conference on Optical Communications and Networks (ICOCN)*, August 23–27, 2021, Qufu, China, p. 1
[31] Gisin N and Thew R 2007 *Nat. Photonics* **1** 165
[32] Andersen U L, Neergaard N J S, van Loock P and Furusawa A 2015 *Nat. Phys.* **11** 713
[33] Amin I, Mishra D, Saini R and Aissa S 2022 *IEEE Commun. Lett.* **26** 31
[34] Sharma P, Agrawal A, Bhatia V, Prakash S and Mishra A K 2021 *IEEE Open J. Commun. Soc.* **2** 2049
[35] Guerrini S, Chiani M and Conti A 2018 2018 *IEEE Globecom Workshops (GC Wkshps)*, December 9–13, 2018, Abu Dhabi, United Arab Emirates, p. 1

[36] Panhwar M A, Khuhro S A, Mazhar T, Deng Z L and Qadir N 2021 *Comput. Sci.* **16** 9
[37] Peev M, Pacher C, Alléaume R, Barreiro C, Bouda J, Boxleitner W, Debuisschert T, Diamanti E, Dianati M and Dynes J 2009 *New J. Phys.* **11** 075001
[38] Cao Y, Zhao Y, Li J, Lin R, Zhang J and Chen J 2021 *IEEE J. Sel. Areas Commun.* **39** 2701
[39] Zhang C, Jing R, Li J, Ma Y, Huo X, Zhao Y and Zhang J 2015 *J. Opt. Commun. Netw.* **7** 1057
[40] McKeown N, Anderson T, Balakrishnan H, Parulkar G, Peterson L, Rexford J, Shenker S and Turner J 2008 *Comput. Commun. Rev.* **38** 69
[41] Schönwälder J, Björklund M and Shafer P 2010 *IEEE Commun. Mag.* **48** 166
[42] Aguado A, Lopez V, Martinez M J, Peev M, Lopez D and Martin V 2018 *J. Opt. Commun. Netw.* **10** 421
[43] Mehic M, Fazio P, Rass S, Maurhart O, Peev M, Poppe A, Rozhon J, Niemiec M and Voznak M 2020 *IEEE ACM Trans. Netw.* **28** 168
[44] Bi L, Miao M and Di X 2023 *Appl. Sci.* **13** 8690
[45] Li M, Quan D and Zhu C 2016 2016 *8th International Conference on Wireless Communications & Signal Processing (WCSP)*, October 13–15, 2016, Yangzhou, China, p. 1
[46] Mehic M, Rass S, Fazio P and Voznak M 2022 *Quantum Key Distribution Networks: A Quality of Service Perspective* (Switzerland: Springer) pp. 109–133