entropy

MDPI

Article

Routing Algorithm Within the Multiple Non-Overlapping Paths' Approach for Quantum Key Distribution Networks

Evgeniy O. Kiktenko, Andrey Tayduganov * and Aleksey K. Fedorov

Laboratory of Quantum Information Technologies, National University of Science and Technology "MISIS", Moscow 119049, Russia

* Correspondence: a.taiduganov@misis.ru

Abstract: We develop a novel key routing algorithm for quantum key distribution (QKD) networks that utilizes a distribution of keys between remote nodes, i.e., not directly connected by a QKD link, through multiple non-overlapping paths. This approach focuses on the security of a QKD network by minimizing potential vulnerabilities associated with individual trusted nodes. The algorithm ensures a balanced allocation of the workload across the QKD network links, while aiming for the target key generation rate between directly connected and remote nodes. We present the results of testing the algorithm on two QKD network models consisting of 6 and 10 nodes. The testing demonstrates the ability of the algorithm to distribute secure keys among the nodes of the network in an all-to-all manner, ensuring that the information-theoretic security of the keys between remote nodes is maintained even when one of the trusted nodes is compromised. These results highlight the potential of the algorithm to improve the performance of QKD networks.

Keywords: quantum communication; quantum key distribution; QKD network; routing scheme

check for updates

Citation: Kiktenko, E.O.; Tayduganov, A.; Fedorov, A.K. Routing Algorithm Within the Multiple Non-Overlapping Paths' Approach for Quantum Key Distribution Networks. *Entropy* **2024**, *26*, 1102. https://doi.org/10.3390/e26121102

Academic Editors: Mohsen Razavi, Masoud Ghalaii, Federico Grasselli and Mirko Pittaluga

Received: 15 November 2024
Revised: 12 December 2024
Accepted: 13 December 2024
Published: 16 December 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Quantum key distribution (QKD) technologies represent an elegant mechanism for solving the key distribution problem [1–3], which is one of the central tasks of cryptography. Existing QKD protocols rely on the use of quantum channels for transmitting information that is encoded in quantum states of single photons (in practice, the transmission of weak laser pulses via fiber-optic and free-space communication channels is used) and authentic classical channels [1]. The idea behind this is that any interception in the process of transmitting quantum information can be detected by monitoring the quantum bit error rate (QBER) in the quantum channel, while authentication of classical communications is needed for the protection from man-in-the-middle attack during post-processing procedures [1,2]. In the view of the fact that the security of QKD is based on the laws of quantum physics, it is guaranteed to be secure against any unforeseen technological developments, such as cryptoanalysis using quantum computing [4]. QKD technologies have attracted a great deal of interest and they are a widely studied field of quantum technologies [3,5]. At the same time, existing realizations of QKD systems still face a number of challenges [5], such as limitations in the key generation rate and distance, as well as practical security.

One of the ways to overcome distance limitations, which are essentially related to losses during the transfer of quantum states, is to use QKD networks [6]. Moreover, QKD networks make key generation services available for multiple users. In the case of quantum communications, standard repeater schemes cannot be used, so quantum repeaters [7–9] are required. Despite significant progress in this field during recent decades (e.g., see Refs. [10–13]), quantum repeaters are not yet widely used in industrial QKD networks.

The most developed concept for large-scale, industrial QKD networks is to use trusted nodes [14–20], which play a role of a user in a QKD protocol that composes a common secret key by using the bit-wise XOR operation for two established keys with neighboring

nodes. The negative side of the trusted-node paradigm is that all the users of the network must trust the node and that it needs to be physically well isolated, while the positive side is a reduction in costs and complexity as compared to the all connected point-to-point links. At this stage, the deployment of QKD networks faces the problem of optimal routing, which has the goal of optimally using resources of a network when establishing keys between two arbitrary users. Existing QKD networks solve the routing problem by means of various protocols [21–32] and tools (such as optical switches [18,33,34]); however, this problem is far from being fully solved, especially in the case of QKD networks of arbitrary topology [26–32].

In this work, we consider the key routing task within the multiple non-overlapping paths' approach [35–39]. The idea of this approach is to minimize the requirement for trust by distributing a key across remote nodes via independent routes. This ensures that the final key remains secure, provided that all trusted nodes on at least one route remain secure. We propose a routing algorithm that is specifically designed for use with the multiple non-overlapping paths' approach. This algorithm differs from previously considered algorithms for single-path routing [21,22,24,29,31] in that it is tailored to handle the unique requirements of the multiple non-overlapping path scenario. The developed algorithm differs from that proposed in [25] by incorporating the limited capacity for key generation in networks and addressing the routing issue for all nodes simultaneously. At the same time, it differs from the more recent scheme for hybrid trusted networks [30] in that all nodes in the network are assumed to have equal trust in it. Importantly, key routing algorithms designed for the standard single-path trusted node schemes do not guarantee the efficiency of a key distribution in the context of the non-overlapping paths' approach. To address this challenge, we propose and test the first key routing algorithm, to our knowledge, which is specifically tailored for the non-overlapping paths' framework.

This paper is organized as follows: In Section 2, we discuss the basic features of the multiple non-overlapping paths' approach. In Section 3, we present the developed routing algorithm. In Section 4, we demonstrate the performance of the algorithm on two QKD network models. Finally, we conclude this article in Section 5.

2. The Concept of the M–Paths' Scheme

To overcome the limitation of the QKD range, multiple short point-to-point QKD links are combined into QKD networks with so-called intermediate trusted nodes [6,40]. In this approach, quantum keys are generated only between the network nodes connected directly by a quantum channel, and trusted nodes perform a key hopping via subsequent bit-wise XOR operations. With the use of this technology, it is possible to build QKD networks of various topologies, like backbone, ring, star or a mixed one.

The downside of the trusted node paradigm is that remote network nodes have to trust the physical integrity of all nodes in the chain between them. It is possible to relax the trust requirement and increase the security by considering an M non-overlapping paths' approach [35–39], where the secret key between two remote nodes is obtained by combining keys from M (with $M > 1$) non-overlapping paths. In the following, we refer to it as the M–paths' approach. We would like to note that, in this context, we refer to a path as an independent key hopping route. From a physical perspective, such independent paths can be realized even within a single transmission line, where certain nodes may be skipped, as is the case of quantum control-based key distribution networks [41,42].

Let us consider a communication between remote nodes i and j via a set of M non-overlapping paths $\mathcal{P}_{ij}^{M}$. Let each path $P \in \mathcal{P}_{ij}^{M}$ provide an ℓ–bit information-theoretically secure key $\mathcal{K}_{ij}^{P}$. Then, the final ℓ–bit secret key to be used for the unconditionally secure data encryption with the one-time pad (OTP) technique can be obtained as follows:

$$\mathcal{K}_{ij} = \bigoplus_{P \in \mathcal{P}_{ij}^{M}} \mathcal{K}_{ij}^{P}, \tag{1}$$

where $\bigoplus$ stands for a bit-wise exclusive or (XOR) operation. Therefore, due to the perfect secrecy of the OTP, the eavesdropping becomes technically more complicated in the M–path scheme since the intermediate nodes from each of M paths must be hacked (compromised) in order to reconstruct the entire secret key. Equivalently, compromising fewer than M nodes from $\mathcal{P}_{ij}^{M}$ would not compromise $\mathcal{K}_{ij}$. Assuming a probability of compromise of a node operated as a trusted node is ϵ_{compr}, we can estimate that the probability of compromising a key between any two remote nodes in the M–paths' regime is upper-bounded by $\epsilon_{\text{compr}}^{M}$.

Here, we would like to emphasize several points related to the M–paths' scheme. First, the applicability of M–path schemes strictly depends on the topology of QKD networks. It is impossible to use this scheme in backbone-type networks, where there is only a single path between two remote nodes, or to apply a three-path scheme within a ring topology of the network graph. Second, distributing an ℓ–bit key using the M–path scheme reduces the remaining keys by ℓ bits for each direct link in each path (we discuss this in more detail in the following section). Therefore, increasing M also increases the "cost" of distributing keys between remote nodes in terms of the remaining key between network nodes. In this regard, the selection of M within the range allowed by the topology should consider a balance between the desired security level, estimated as $\epsilon_{\text{compr}}^{M}$, and the increased key usage in the network. Thirdly, the secret key $\mathcal{K}_{ij}$ can be obtained by concatenating several keys corresponding to different M–element sets of non-overlapping paths $\{\mathcal{P}_{ij}^{M(k)}\}_k$:

$$\mathcal{K}_{ij} = \Big(\bigoplus_{P \in \mathcal{P}_{ij}^{M(1)}} \mathcal{K}_{ij}^{P} \Big) \| \Big(\bigoplus_{P' \in \mathcal{P}_{ij}^{M(2)}} \mathcal{K}_{ij}^{P'} \Big) \| \dots , \tag{2}$$

where $\|$ denotes concatenation. It should be noted that some of the paths in the sets $\{\mathcal{P}_{ij}^{M(k)}\}_k$ may have common elements. Additionally, the lengths of concatenated parts may differ. The construction of $\mathcal{K}_{ij}$ in the form of Equation (2), assuming it is permitted by the network topology, enables a more balanced distribution of key consumption across the network. The last point raises an important challenge in finding a practical and efficient key routing algorithm within the M–path approach for QKD networks of arbitrary topology. This algorithm should allow for the selection of suitable combinations of paths in order to distribute keys between pairs of remote nodes. Additionally, it should ensure a sufficient reserve of available keys for directly connected nodes. We consider the construction of such an algorithm in the section below.

3. Routing Algorithm for the M–Path Scheme

We begin with introducing some formal definitions and notations used later. An arbitrary QKD network with N nodes is represented as a graph $G = (V, E)$, where the vertices $V = \{i\}_{i=0}^{N-1}$ are the network nodes and the edges $E \subseteq \{(i,j) \,|\, i,j \in V\}$ are the fiber optic lines connecting them. The edge weight corresponds to the secret key generation rate of the corresponding section of the fiber optic line. The definition of the M–path routing scheme obviously implies that the degree of each vertex, i.e., the number of edges that are incident to the vertex, is required to be $\deg(i) \geq M$. A path in a graph is a finite sequence of edges which joins a sequence of distinct vertices. An open path from Node i to Node j can be defined in a short way as a sequence of adjacent vertices, connected by edges,

$$P_{ij} \coloneqq (i, k, l, \dots, m, j) , \tag{3}$$

with $(i,k), (k,l), \dots, (m,j) \in E$.

Let $R_{ij} = R_{ji}$ be the secret key generation rate for a pairwise QKD link or chain of links connecting nodes i and j such that $R_{ij} > 0$ if $(i,j) \in E$ is zero. The average length of the local secret key, K_{ij}, accumulated during some time period τ, is equal to $|K_{ij}| = R_{ij}\tau$. Let T_{ij} be a target secret key generation rate between the nodes i and j, which can be associated

with the key consumption rate. Note that it is possible to have $T_{ij} > 0$ even if $(i,j) \notin E$. Then, the goal is to develop a (sub)optimal key flow that provides the desired T_{ij} for all pairs (i,j) by manipulating the effective key rates R_{ij}^{eff}.

To illustrate the concept of the effective rate, consider a simple network of five nodes in Figure 1 and set $M = 2$. The communication between the remote nodes 0 and 4 ($R_{04} \equiv 0$) in the 2–path scheme can be completed via three possible pairs of non-overlapping paths,

$$\begin{aligned} \mathcal{P}_{04}^{2(1)} &= \{(0,1,4),(0,2,4)\}\,, \\ \mathcal{P}_{04}^{2(2)} &= \{(0,2,4),(0,3,4)\}\,, \\ \mathcal{P}_{04}^{2(3)} &= \{(0,1,4),(0,3,4)\}\,, \end{aligned} \tag{4}$$

while the communication between nodes 1 and 3 ($R_{13} \equiv 0$) is feasible only via one pair,

$$\mathcal{P}_{13}^{2} = \{(1,0,3),(1,4,3)\}\,. \tag{5}$$

Every local accumulated secret key K_{ij} is split into subsamples in the following way:

$$\begin{aligned} K_{01} &= K_{01}^{\text{eff}} \| \delta K_{01}^{(0,1,4)} \| \delta K_{01}^{(0,1,4)\prime} \| \delta K_{01}^{(1,0,3)}\,, \\ K_{02} &= K_{02}^{\text{eff}} \| \delta K_{02}^{(0,2,4)} \| \delta K_{02}^{(0,2,4)\prime}\,, \\ K_{03} &= K_{03}^{\text{eff}} \| \delta K_{03}^{(0,3,4)} \| \delta K_{03}^{(0,3,4)\prime} \| \delta K_{03}^{(1,0,3)}\,, \\ K_{14} &= K_{14}^{\text{eff}} \| \delta K_{14}^{(0,1,4)} \| \delta K_{14}^{(0,1,4)\prime} \| \delta K_{14}^{(1,4,3)}\,, \\ K_{24} &= K_{24}^{\text{eff}} \| \delta K_{24}^{(0,2,4)} \| \delta K_{24}^{(0,2,4)\prime}\,, \\ K_{34} &= K_{34}^{\text{eff}} \| \delta K_{34}^{(0,3,4)} \| \delta K_{34}^{(0,3,4)\prime} \| \delta K_{34}^{(1,4,3)}\,, \end{aligned} \tag{6}$$

where the keys $\delta K_{ij}^{P(\prime)}$ are sacrificed for the key hopping through the path P with the OTP technique, while the remaining effective keys K_{ij}^{eff} are used for direct communication between connected nodes. We note that $\delta K_{ij}^{P} \neq \delta K_{ij}^{P\prime}$ since the path belongs to different M–path sets (see Equation (4)). The OTP technique requires the key lengths to satisfy the following conditions:

$$\begin{aligned} &|\delta K_{01}^{(0,1,4)}| = |\delta K_{14}^{(0,1,4)}| = |\delta K_{02}^{(0,2,4)}| = |\delta K_{24}^{(0,2,4)}|\,, \\ &|\delta K_{02}^{(0,2,4)\prime}| = |\delta K_{24}^{(0,2,4)\prime}| = |\delta K_{03}^{(0,3,4)}| = |\delta K_{34}^{(0,3,4)}|\,, \\ &|\delta K_{01}^{(0,1,4)\prime}| = |\delta K_{14}^{(0,1,4)\prime}| = |\delta K_{03}^{(0,3,4)\prime}| = |\delta K_{34}^{(0,3,4)\prime}|\,, \\ &|\delta K_{01}^{(1,0,3)}| = |\delta K_{03}^{(1,0,3)}| = |\delta K_{14}^{(1,4,3)}| = |\delta K_{34}^{(1,4,3)}|\,. \end{aligned}$$

Then, the final effective keys to encrypt the communication between remote nodes are formed according to Equation (2):

$$\begin{aligned} K_{04}^{\text{eff}} = &\left(\delta K_{01}^{(0,1,4)} \oplus \delta K_{02}^{(0,2,4)}\right) \\ &\| \left(\delta K_{02}^{(0,2,4)\prime} \oplus \delta K_{03}^{(0,3,4)}\right) \\ &\| \left(\delta K_{01}^{(0,1,4)\prime} \oplus \delta K_{03}^{(0,3,4)\prime}\right)\,, \\ K_{13}^{\text{eff}} = &\,\delta K_{01}^{(1,0,3)} \oplus \delta K_{14}^{(1,4,3)}\,. \end{aligned} \tag{7}$$

Now, having all nodes "connected", we can define the effective generation rate for arbitrary pair of nodes in the network as

$$R_{ij}^{\text{eff}} := \frac{|K_{ij}^{\text{eff}}|}{\tau} > 0\,. \tag{8}$$

It is easy to see that $R_{ij}^{\text{eff}} < R_{ij}$ for all $(i,j) \in E$. In this way, the key management task is to find an optimal set of $\delta K_{ij}^{P(\prime)}$ in Equation (6) such that $R_{ij}^{\text{eff}} \geq T_{ij}$ for all $i,j \in V$.

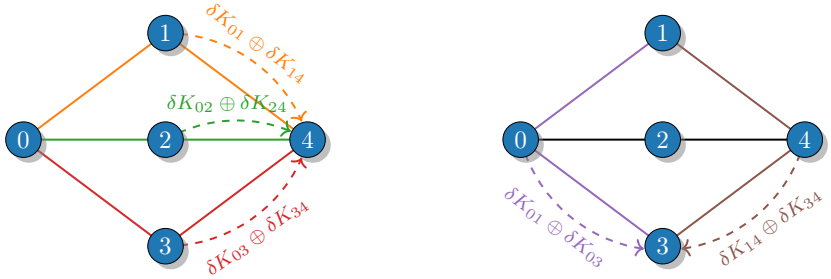

Figure 1. The key distribution between remote nodes 0 and 4 (**left**) and 1 and 3 (**right**) via various non-overlapping paths for a network of five nodes. The path index of δK_{ij} is omitted for simplicity.

Here, we would like to draw attention to the fact that all OTP-secured communications must be authenticated, which also requires the use of a secret key. Due to the fixed consumption of secret keys in the authentication of each message, it is more efficient in practice to increase the interval between "key distribution rounds" and, consequently, the length of the keys transmitted in each message. This approach helps to minimize the relative costs associated with authentication. In the following section, we assume that the key consumption for authentication can be neglected compared to the transmitted key material.

The proposed routing Algorithm 1 follows an iterative process (see an example of a step-by-step workflow in Appendix A). As the basic target quantity to be minimized in each iteration, we consider

$$\Delta := \max_{i,j \in V} \left[T_{ij} - R_{ij}^{\text{eff}} \right], \tag{9}$$

which provides the discrepancy between the target and effective rates for the "worst" pair of nodes. Having $\Delta \leq 0$ means that the goal is achieved, and thus the algorithm can be stopped. At every iteration $r = 1, 2, \ldots, r_{\max}$, where the hyperparameter $r_{\max}$ stands for the maximal allowed number of iterations, we first search for the currently "worst" pair of nodes in the graph (i.e., with maximum value of Δ), denoted by (i^*, j^*). If the found pair is connected directly, then the algorithm stops. In the case of multiple pair candidates, a random candidate is selected. Then, among all possible sets of M non-overlapping paths from i^* to j^*, denoted by $\mathcal{L}_{i^*j^*}^{M}$, the optimal set $\mathcal{P}_{i^*j^*}^{M,\text{opt}}$ is selected by searching for a set that has a path containing a pair of connected nodes with the minimal rate deficiency. In Algorithm 1, we formalize it by introducing the deficiency function $\mathbf{D}(\cdot)$, which returns the maximal deficiency among all direct links within an input set of M non-overlapping paths. Again, if multiple candidates are found, a random candidate over candidates with the minimal total length is selected (this is formalized by the `rand_ch_min_dist`$(\cdot)$ function). When the optimal M–path combination is determined, $R_{i^*j^*}^{\text{eff}}$ is increased by a small amount of δR (which is an input hyperparameter of the algorithm), while R_{ij}^{eff} is decreased by δR for all adjacent nodes from every path in $\mathcal{P}_{i^*j^*}^{M,\text{opt}}$.

After that, we calculate the new value of the cost function according to Equation (9) and store it in Δ_{upd}. If the new value is "worse" than the one in the beginning of the iteration ($\Delta_{\text{upd}} > \Delta$), then the algorithm stops; otherwise, the information about the obtained key routing path $\mathcal{P}_{i^*j^*}^{M,\text{opt}}$ between i^* and j^* is added to the list routing_list, and the algorithm proceeds to the next iteration. Here, we assume that routing_list consists of records of the form $(\mathcal{P}_{i'j'}^{M} : R')$. Each of these records means that during some fixed time period τ, a key of length $R'\tau$ has to be distributed between i' and j' over M paths in $\mathcal{P}_{i'j'}^{M}$. Thus, if

routing_list already contains a record of the form $(\mathcal{P}_{i^*j^*}^{M,\mathrm{opt}} : R_{i^*j^*}^{\mathrm{cur}})$, then it is replaced by $(\mathcal{P}_{i^*j^*}^{M,\mathrm{opt}} : R_{i^*j^*}^{\mathrm{cur}} + \delta R)$, or, otherwise, the new record $(\mathcal{P}_{i^*j^*}^{M,\mathrm{opt}} : \delta R)$ is added.

Algorithm 1

Require: (R_{ij}), (T_{ij}), δR, $r_{\max}$
Ensure: routing_list
  $R_{ij}^{\mathrm{eff}} = R_{ij} \ \ \forall i, j \in V$ ▷ initialization of the effective key generation matrix
  $r = 0$ ▷ initialization of iteration counter
  routing_list = [ ] ▷ initialization of empty routing list
  $\Delta = \max_{i,j \in V}[T_{ij} - R_{ij}]$
  while $\Delta > 0 \wedge r < r_{\max}$ **do**
    $D(i,j) \coloneqq T_{ij} - R_{ij}^{\mathrm{eff}}$ ▷ current rate deficiency function
    $(i^*, j^*) = \mathtt{rand\_ch}\{\arg\max_{i,j \in V} D(i,j)\}$ ▷ selecting a random pair over pairs with the "worst" deficiency
    if $(i^*, j^*) \in E$ **then**
      return routing_list ▷ stopping the algorithm
    end if
    $\mathcal{P}_{i^*j^*} = \{P = (i^*, \ldots, j^*)\}$ ▷ set of all possible paths from i^* to j^*
    $\mathcal{L}_{i^*j^*}^{M} = \{\mathcal{P}_{i^*j^*}^{M} = \{P \in \mathcal{P}_{i^*j^*} : P \cap P' = \{i^*, j^*\} \ \forall P' \in \mathcal{P}_{i^*j^*}^{M}, P' \neq P\} : |\mathcal{P}_{i^*j^*}^{M}| = M\}$ ▷ list of all possible combinations of M non-overlapping paths from i^* to j^*
    $\mathbf{D}(\mathcal{P}_{i^*j^*}^{M}) \coloneqq \max_{P \in \mathcal{P}_{i^*j^*}^{M}} \max_{(i,j) \in P} D(i,j)$ ▷ deficiency function which characterizes the M–path combination capacity ("deficiency of the worst link in the worst path")
    $\mathcal{P}_{i^*j^*}^{M,\text{pre-opt}} = \arg\min_{\mathcal{P}_{i^*j^*}^{M} \in \mathcal{L}} \mathbf{D}(\mathcal{P}_{i^*j^*}^{M})$
    $\mathcal{P}_{i^*j^*}^{M,\mathrm{opt}} = \mathtt{rand\_ch\_min\_dist}\{\mathcal{P}_{i^*j^*}^{M,\text{pre-opt}}\}$ ▷ taking the optimal M–path combination
    $R_{i^*j^*}^{\mathrm{eff}} = R_{i^*j^*}^{\mathrm{eff}} + \delta R$ ▷ increase the effective key rate for the "worst" pair
    for all $P \in \mathcal{P}_{i^*j^*}^{M,\mathrm{opt}}$ **do**
      for all $i, j \in P : (i,j) \in E$ **do**
        $R_{ij}^{\mathrm{eff}} = R_{ij}^{\mathrm{eff}} - \delta R$ ▷ decrease the effective key rates for adjacent pairs
      end for
    end for
    $\Delta_{\mathrm{upd}} = \max_{i,j \in V}[T_{ij} - R_{ij}^{\mathrm{eff}}]$
    if $\Delta_{\mathrm{upd}} \leq \Delta$ **then**
      update routing_list with a record $(\mathcal{P}_{i^*j^*}^{M,\mathrm{opt}} : \delta R)$
      $\Delta = \Delta_{\mathrm{upd}}$ ▷ updating the value if the cost and function
      $r = r + 1$ ▷ incrementing the counter
    else
      return routing_list ▷ stopping the algorithm
    end if
  end while
  return routing_list

Finally, we would like to discuss the role of two hyperparameters: $r_{\max}$ and δR. By bounding the number of iterations, $r_{\max}$ limits the maximum running time of the algorithm. One can also consider replacing the condition $r < r_{\max}$ in the main loop of the algorithm by a condition that checks if the current running time is less than the maximum allowed time. If there are no limits on running time, $r_{\max}$ can be set to infinity. δR determines the size of the step at each iteration. A smaller δR results in more iterations and a higher precision in the final value of the cost function (Δ). As we will see in the testing in the next section, reducing δR can improve the cost function by the amount of δR. However, from a practical perspective, it is not advisable to reduce δR below a level corresponding to the typical cost of a secret key for authentication purposes, assumed to be negligible within the construction of the algorithm.

4. Performance Demonstration of the Algorithm

To demonstrate the performance of the developed algorithm, we consider the task of routing the key within the 2–path scheme for two examples of QKD networks.

The first QKD network is represented by a six-vertex graph shown in Figure 2a. For simplicity, we assume that all adjacent links are equidistantly located and have equal initial key generation rates, namely $R_{ij} := 1\,\text{kbit/s}$ for all $(i,j) \in E$. The target rates are also set to be equal, $T_{ij} := 0.1\,\text{kbit/s}$, for all possible pairs. The decrease in the cost function Δ with iteration number r is shown in Figure 2b. The algorithm completes the task after 80, 160 and 800 iterations, respectively. For all considered values of δR, the matrix of final effective rates R^{eff} approaches to a form shown in Figure 2c. The corresponding output routing list for remote nodes is obtained in the following:

$$\begin{aligned} \text{routing\_list} = [&\{(0,1,2),(0,3,2)\} : 0.1\,,\\ &\{(1,0,3),(1,2,3)\} : 0.1\,,\\ &\{(1,2,5),(1,4,5)\} : 0.1\,,\\ &\{(2,1,4),(2,5,4)\} : 0.1\,,\\ &\{(0,1,4,5),(0,3,2,5)\} : 0.1\,,\\ &\{(3,0,1,4),(3,2,5,4)\} : 0.1\,,\\ &\{(0,1,4),(0,3,2,5,4)\} : 0.1\,,\\ &\{(3,0,1,4,5),(3,2,5)\} : 0.1]\,. \end{aligned} \tag{10}$$

It can be seen that the resulting paths follow the graph symmetry and match intuitive expectations. It should be noted that each pair of remote nodes (i,j) has a unique pair of paths through which the key is distributed between them. Considering the key management, the accumulated secret key, e.g., K_{01}, is split into multiple subsamples: six keys $\{\delta K_{01}^P\}$ of equal length $|\delta K_{01}^P| = 0.1|K_{01}|$ are used for the key hopping via the paths (0,1,2), (1,0,3), (0,1,4,5), (3,0,1,4), (0,1,4) and (3,0,1,4,5), while the remaining key $K_{01}^{\text{eff}} = K_{01} \setminus (\bigcup_P \delta K_{01}^P)$ of length $|K_{01}^{\text{eff}}| = 0.4|K_{01}|$ can be entirely used for the direct communication between nodes 0 and 1. A similar key splitting procedure according to the R^{eff} matrix in Figure 2c is performed for other edges of the graph. Then, the effective secret key to encrypt the communication between a pair of remote nodes, e.g., 1 and 3, is $K_{13}^{\text{eff}} = \delta K_{01}^{(1,0,3)} \oplus \delta K_{12}^{(1,2,3)}$.

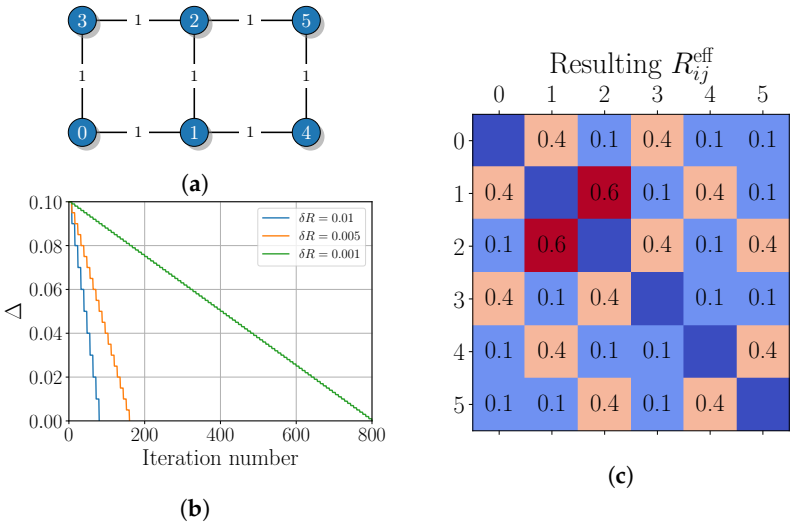

Figure 2. (**a**) The network graph with edges, labeled with secret key generation rates in kbps. (**b**) The iteration number dependence of the cost function. (**c**) The resulting effective secret key generation rates for various pairs of network nodes.

The second considered example of a more complex QKD network topology, which is taken from ref. [29], is shown in Figure 3a. In contrast to the previous case, the network has different rates for each link and does not possess a symmetrical structure. The secret key generation rates are simulated for the decoy-state BB84 protocol [43–47] using the link distances from the corresponding graph in ref. [29]. For this example, we set $T_{ij} := 1.0\,\mathrm{kbit/s}$ for all possible pairs of nodes and $M = 2$. The behavior of the cost function Δ with iteration number r for $\delta R \in \{0.1, 0.05, 0.01\}$ is shown in Figure 3b. The algorithm stops after 78, 158, and 793 iterations, respectively. One can notice that Δ does not approach to zero like in the first example and stops around 0.75 since the chosen target matrix turns out to be overrated for this scheme. We also note that the choice of δR slightly affects the resulting value of the cost function.

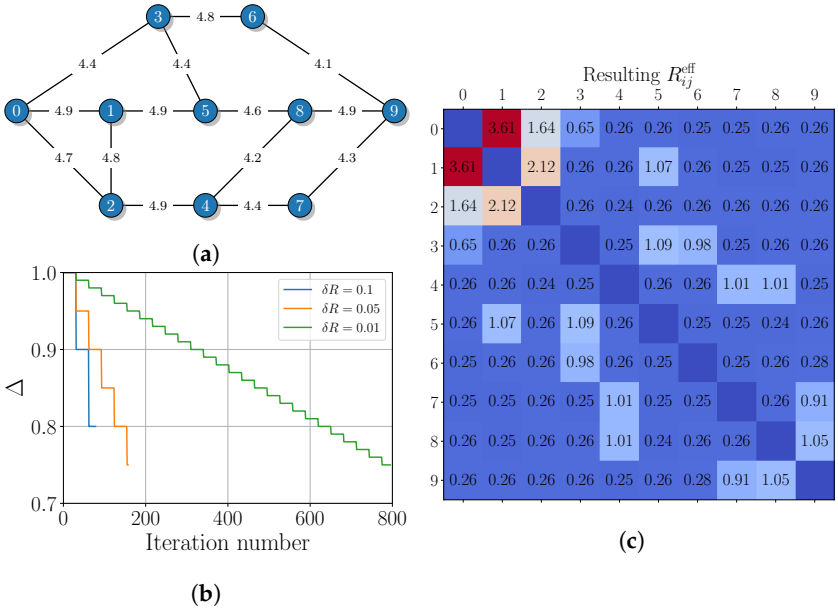

Figure 3. (**a**) The network graph from ref. [29] with edges that are labeled with the secret key generation rates in kbps, simulated for the decoy-state BB84 protocol using the simple theoretical model from ref. [44] and taking into account the detector's dead time effect. (**b**) The iteration number dependence of the cost function. (**c**) The resulting effective secret key generation rates (obtained with $\delta R = 0.01$) for various pairs of network nodes.

The resulting form of the R^{eff} matrix for $\delta R = 0.01$, shown in Figure 3c, demonstrates that the algorithm provides a close-to-uniform distribution of the effective key generation rates among remote nodes. The 11 links, which keep an effective key generation rate noticeably above a "water level" of 0.25, correspond to directly connected nodes. However, note that the key generation in direct links $(2,4)$, $(5,8)$, and $(6,9)$ drops to the "water level", stopping the algorithm.

For an additional illustration, we present all the path pairs from routing_list used for the key distribution between the 0th and 9th nodes:

$$\begin{aligned} &(0,2,4,7,9),(0,3,5,8,9) : 0.02\,,\\ &(0,1,5,8,9),(0,2,4,7,9) : 0.03\,,\\ &(0,1,5,8,9),(0,3,6,9) : 0.1\,,\\ &(0,2,4,7,9),(0,3,6,9) : 0.1\,,\\ &(0,2,4,8,9),(0,3,6,9) : 0.01\,. \end{aligned} \tag{11}$$

Notably, the effective key generation rate of $R_{09}^{\mathrm{eff}} = 0.26$ is realized in a quite non-trivial way via five different non-overlapping path pairs.

Then, the final effective secret key between nodes 0 and 9, assembled according to Equation (2), can be written as

$$\begin{aligned} K_{09}^{\text{eff}} = &\left(\delta K_{02}^{(0,2,4,7,9)} \oplus \delta K_{03}^{(0,3,5,8,9)}\right) \\ &\| \left(\delta K_{01}^{(0,1,5,8,9)} \oplus \delta K_{02}^{(0,2,4,7,9)\prime}\right) \\ &\| \left(\delta K_{01}^{(0,1,5,8,9)\prime} \oplus \delta K_{03}^{(0,3,6,9)}\right) \\ &\| \left(\delta K_{02}^{(0,2,4,7,9)\prime\prime} \oplus \delta K_{03}^{(0,3,6,9)\prime}\right) \\ &\| \left(\delta K_{02}^{(0,2,4,8,9)} \oplus \delta K_{03}^{(0,3,6,9)\prime\prime}\right), \end{aligned} \tag{12}$$

where the lengths of the XORed pairs of keys depend on the respective effective rates. A more detailed description of such key management becomes highly non-trivial in this case and is beyond this research.

5. Conclusions and Outlook

In this work, we have addressed the issue of trust reduction in QKD networks by studying the key distribution among remote nodes through multiple non-overlapping paths (M–path scheme). We have proposed a novel iterative greedy algorithm for key routing in QKD networks, aimed at effectively facilitating the key distribution between remote nodes while utilizing the M–path scheme. The efficiency of the proposed algorithm has been shown through case studies involving two QKD networks comprising nodes 6 and 10, respectively. Our results have demonstrated the potential for generating non-trivial key routing paths between remote nodes while achieving a balanced load distribution across directly connected network nodes. This work not only contributes to enhancing the reliability of QKD systems but also opens avenues for further research in optimizing key distribution strategies for QKD networks.

As a potential limitation on the applicability of the developed algorithm, we would like to emphasize its strong dependence on network topology. Specifically, it is assumed that any two remote nodes can be connected by at least M non-overlapping paths. However, in a real QKD network, this assumption may not hold. It is possible to envision a scenario where some pairs of remote nodes are connected by non-overlapping paths, but others are not. This limitation can be addressed by considering a more flexible approach with different requirements for the value of M for each pair, or by considering "partially overlapping path" scenarios. This appears to be a promising avenue for future research.

Author Contributions: A.K.F. and E.O.K. formulated the problem statement and goals of this study; E.O.K. originated the initial concept for the algorithm and created a prototype for its implementation; A.T. refined the details of the algorithm and devised test cases for its validation; A.T. and E.O.K. drafted the initial version of this paper, with a contribution from A.K.F.; A.K.F. supervised the project. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Priority 2030 program at the National University of Science and Technology "MISIS" under the project K1-2022-027.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Acknowledgments: We would like to thank V. Dubinin for their discussions.

Conflicts of Interest: Owing to the employments and consulting activities, the authors have financial interests in the commercial applications of QKD technologies. The authors declare no non-financial conflicts of interest.

Appendix A

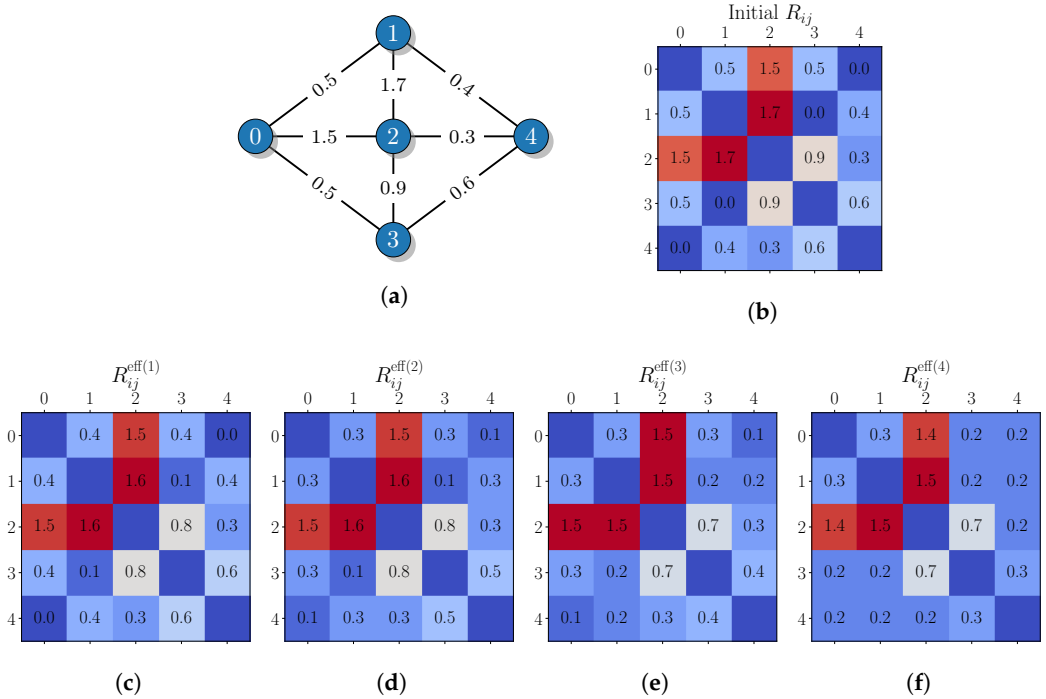

Figure A1. (**a**) The network graph with edges, labeled with secret key generation rates. (**b**) The initial key rate matrix. (**c**–**f**) The resulting effective secret key generation rates after every iteration.

To illustrate a step-by-step workflow of Algorithm 1, we consider a simple network of five nodes, depicted in Figure A1a. The graph edges are labeled with corresponding secret key generation rates, which are represented by the initial key rate matrix in Figure A1b. The target rates are assumed to be $T_{ij} = 0.2$. The iteration step is set to be $\delta R = 0.1$.

1. At the start, the remote nodes apparently have the "worst" node pair deficiency,

$$\arg\max_{i,j \in V} D(i,j) = \{(0,4),(1,3)\}\,.$$

Hence, a random candidate, e.g., $(1,3)$, is chosen. Then, the list of all two non-overlapping path combinations from Node 1 to Node 3 is constructed,

$$\begin{aligned}\mathcal{L}_{13}^{2} = [&\{(1,0,2,3),(1,4,3)\}\,, && \mathbf{D} = -0.2\\ &\{(1,0,3),(1,4,2,3)\}\,, && \mathbf{D} = -0.1\\ &\{(1,0,3),(1,4,3)\}\,, && \mathbf{D} = -0.2\\ &\{(1,0,3),(1,2,4,3)\}\,, && \mathbf{D} = -0.1\\ &\{(1,0,3),(1,2,3)\}\,, && \mathbf{D} = -0.3\\ &\{(1,4,3),(1,2,0,3)\}\,, && \mathbf{D} = -0.2\\ &\{(1,4,3),(1,2,3)]\}\,. && \mathbf{D} = -0.2\end{aligned}$$

Here, in the right column, we put values of $\mathbf{D}$ for each combination of paths. It is computed as a maximal deficiency (target rate minus current rate) over all links in both paths. For example, for $\{(1,0,3),(1,4,3)\}$, we have $\mathbf{D} = \max(0.2-0.5, 0.2-0.5, 0.2-0.4, 0.2-0.6) = -0.2$. The lowest value of the path deficiency function, $\mathbf{D} = -0.3$, determines the optimal two-path combination,

$$\mathcal{P}_{13}^{2,\text{opt}} = \{(1,0,3),(1,2,3)\}\,.$$

The effective rates for all node pairs from $\mathcal{P}_{13}^{2,\mathrm{opt}}$ are computed as follows:

$$R_{13}^{\mathrm{eff}} = \delta R\,, \quad R_{ij}^{\mathrm{eff}} = R_{ij} - \delta R \;\; \forall ij \in \{01, 03, 12, 23\}\,.$$

The result can be seen in Figure A1c. The routing list, which was initially empty, is appended with the first entry:

$$\mathsf{routing\_list} = [\{(1,0,3),(1,2,3)\} : 0.1]. \tag{A1}$$

2. After the first iteration, $(0,4)$ automatically becomes the "worst" pair, for which the list of non-overlapping paths with corresponding two-path deficiencies is

$$\begin{aligned}
\mathcal{L}_{04}^{2} = [&\{(0,1,4),(0,2,4)\}\,, && \mathbf{D} = -0.1\\
&\{(0,1,4),(0,2,3,4)\}\,, && \mathbf{D} = -0.2\\
&\{(0,1,4),(0,3,4)\}\,, && \mathbf{D} = -0.2\\
&\{(0,1,4),(0,3,2,4)\}\,, && \mathbf{D} = -0.1\\
&\{(0,1,2,4),(0,3,4)\}\,, && \mathbf{D} = -0.1\\
&\{(0,2,4),(0,3,4)\}\,, && \mathbf{D} = -0.1\\
&\{(0,2,1,4),(0,3,4)]\,, && \mathbf{D} = -0.2
\end{aligned}$$

One can see that there are three candidates with a lowest value of $\mathbf{D} = -0.2$, which form the set $\mathcal{P}_{04}^{2,\text{pre-opt}}$, but only one has the lowest distance (i.e., the total number of edges that form two paths), equal to 4. Hence,

$$\mathcal{P}_{04}^{2,\mathrm{opt}} = \{(0,1,4),(0,3,4)\}\,.$$

The effective rates for all node pairs from $\mathcal{P}_{04}^{2,\mathrm{opt}}$ are recomputed as follows:

$$R_{04}^{\mathrm{eff}} = \delta R\,, \quad R_{01(03)}^{\mathrm{eff}} \to R_{01(03)}^{\mathrm{eff}} - \delta R\,, \quad R_{14(34)}^{\mathrm{eff}} = R_{14(34)} - \delta R\,.$$

The result can be seen in Figure A1d. The routing list is appended with the second entry:

$$\begin{aligned}
\mathsf{routing_list} = [&\{(1,0,3),(1,2,3)\} : 0.1\,,\\
&\{(0,1,4),(0,3,4)\} : 0.1]\,.
\end{aligned} \tag{A2}$$

3. After the second iteration, there are again two candidates with a "worst" node pair deficiency, $\arg\max_{i,j\in V} D(i,j) = \{(0,4),(1,3)\}$, among which $(1,3)$ is chosen, for example. The non-overlapping path list is apparently the same as for the first iteration; however, the deficiencies are renewed as follows:

$$\begin{aligned}
\mathcal{L}_{13}^{2} = [&\{(1,0,2,3),(1,4,3)\}\,, && \mathbf{D} = -0.1\\
&\{(1,0,3),(1,4,2,3)\}\,, && \mathbf{D} = -0.1\\
&\{(1,0,3),(1,4,3)\}\,, && \mathbf{D} = -0.1\\
&\{(1,0,3),(1,2,4,3)\}\,, && \mathbf{D} = -0.1\\
&\{(1,0,3),(1,2,3)\}\,, && \mathbf{D} = -0.1\\
&\{(1,4,3),(1,2,0,3)\}\,, && \mathbf{D} = -0.1\\
&\{(1,4,3),(1,2,3)]\,, && \mathbf{D} = -0.1
\end{aligned}$$

Among three candidates with the lowest distance, a random one is chosen as follows:

$$\mathcal{P}_{13}^{2,\mathrm{opt}} = \{(1,4,3),(1,2,3)\}\,.$$

The new effective rates are

$$R_{13}^{\text{eff}} \to R_{13}^{\text{eff}} + \delta R\,, \quad R_{ij}^{\text{eff}} \to R_{ij}^{\text{eff}} - \delta R \;\; \forall ij \in \{14, 34, 12, 23\}\,.$$

The result can be seen in Figure A1e. The routing list is appended with a new entry, as follows:

$$\begin{aligned} \text{routing\_list} = [&\{(1,0,3),(1,2,3)\} : 0.1\,, \\ &\{(1,4,3),(1,2,3)\} : 0.1\,, \\ &\{(0,1,4),(0,3,4)\} : 0.1]\,. \end{aligned} \tag{A3}$$

4. After the third iteration, $D(0,4) = 0.1$ turns out to be the highest. Among all paths from Node 0 to Node 4,

$$\begin{aligned} \mathcal{L}_{04}^{2} = [&\{(0,1,4),(0,2,4)\}\,, && \mathbf{D} = 0 \\ &\{(0,1,4),(0,2,3,4)\}\,, && \mathbf{D} = 0 \\ &\{(0,1,4),(0,3,4)\}\,, && \mathbf{D} = 0 \\ &\{(0,1,4),(0,3,2,4)\}\,, && \mathbf{D} = 0 \\ &\{(0,1,2,4),(0,3,4)\}\,, && \mathbf{D} = -0.1 \\ &\{(0,2,4),(0,3,4)\}\,, && \mathbf{D} = -0.1 \\ &\{(0,2,1,4),(0,3,4)]\,, && \mathbf{D} = 0 \end{aligned}$$

there are two candidates with the lowest two-path deficiency $\mathbf{D} = -0.1$. The optimal path combination with the lowest distance is

$$\mathcal{P}_{04}^{2,\text{opt}} = \{(0,2,4),(0,3,4)\}\,,$$

In this way, the final rate correction is

$$R_{04}^{\text{eff}} \to R_{04}^{\text{eff}} + \delta R\,, \quad R_{ij}^{\text{eff}} \to R_{ij}^{\text{eff}} - \delta R \;\; \forall ij \in \{02, 24, 03, 34\}\,. \tag{A4}$$

The result can be seen in Figure A1e. The resulting routing list takes the following form:

$$\begin{aligned} \text{routing\_list} = [&\{(1,0,3),(1,2,3)\} : 0.1\,, \\ &\{(1,4,3),(1,2,3)\} : 0.1\,, \\ &\{(0,1,4),(0,3,4)\} : 0.1\,, \\ &\{(0,2,4),(0,3,4)\} : 0.1]\,. \end{aligned} \tag{A5}$$

Since all effective rates are not lower than the target rate, the algorithm stops.

References

1. Gisin, N.; Ribordy, G.; Tittel, W.; Zbinden, H. Quantum cryptography. *Rev. Mod. Phys.* **2002**, *74*, 145–195. [CrossRef]
2. Scarani, V.; Bechmann-Pasquinucci, H.; Cerf, N.J.; Dušek, M.; Lütkenhaus, N.; Peev, M. The security of practical quantum key distribution. *Rev. Mod. Phys.* **2009**, *81*, 1301–1350. [CrossRef]
3. Lo, H.K.; Curty, M.; Tamaki, K. Secure quantum key distribution. *Nat. Photonics* **2014**, *8*, 595–604. [CrossRef]
4. Shor, P.W. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *SIAM Rev.* **1999**, *41*, 303–332. [CrossRef]
5. Diamanti, E.; Lo, H.K.; Qi, B.; Yuan, Z. Practical challenges in quantum key distribution. *npj Quantum Inf.* **2016**, *2*, 16025. [CrossRef]
6. Zhang, Q.; Xu, F.; Chen, Y.A.; Peng, C.Z.; Pan, J.W. Large scale quantum key distribution: Challenges and solutions. *Opt. Express* **2018**, *26*, 24260–24273. [CrossRef]
7. Briegel, H.J.; Dür, W.; Cirac, J.I.; Zoller, P. Quantum Repeaters: The Role of Imperfect Local Operations in Quantum Communication. *Phys. Rev. Lett.* **1998**, *81*, 5932–5935. [CrossRef]
8. Duan, L.M.; Lukin, M.D.; Cirac, J.I.; Zoller, P. Long-distance quantum communication with atomic ensembles and linear optics. *Nature* **2001**, *414*, 413–418. [CrossRef] [PubMed]

9. Sangouard, N.; Simon, C.; de Riedmatten, H.; Gisin, N. Quantum repeaters based on atomic ensembles and linear optics. *Rev. Mod. Phys.* **2011**, *83*, 33–80. [CrossRef]
10. Bhaskar, M.K.; Riedinger, R.; Machielse, B.; Levonian, D.S.; Nguyen, C.T.; Knall, E.N.; Park, H.; Englund, D.; Lončar, M.; Sukachev, D.D.; et al. Experimental demonstration of memory-enhanced quantum communication. *Nature* **2020**, *580*, 60–64. [CrossRef]
11. Holzäpfel, A.; Etesse, J.; Kaczmarek, K.T.; Tiranov, A.; Gisin, N.; Afzelius, M. Optical storage for 0.53 s in a solid-state atomic frequency comb memory using dynamical decoupling. *New J. Phys.* **2020**, *22*, 063009. [CrossRef]
12. Bersin, E.; Sutula, M.; Huan, Y.Q.; Suleymanzade, A.; Assumpcao, D.R.; Wei, Y.C.; Stas, P.J.; Knaut, C.M.; Knall, E.N.; Langrock, C.; et al. Telecom Networking with a Diamond Quantum Memory. *PRX Quantum* **2024**, *5*, 010303. [CrossRef]
13. Knaut, C.M.; Suleymanzade, A.; Wei, Y.C.; Assumpcao, D.R.; Stas, P.J.; Huan, Y.Q.; Machielse, B.; Knall, E.N.; Sutula, M.; Baranes, G.; et al. Entanglement of nanophotonic quantum memory nodes in a telecom network. *Nature* **2024**, *629*, 573–578. [CrossRef]
14. Elliott, C. The DARPA Quantum Network. *arXiv* **2004**, arXiv:quant-ph/0412029.
15. Peev, M.; Pacher, C.; Alléaume, R.; Barreiro, C.; Bouda, J.; Boxleitner, W.; Debuisschert, T.; Diamanti, E.; Dianati, M.; Dynes, J.F.; et al. The SECOQC quantum key distribution network in Vienna. *New J. Phys.* **2009**, *11*, 075001. [CrossRef]
16. Stucki, D.; Legré, M.; Buntschu, F.; Clausen, B.; Felber, N.; Gisin, N.; Henzen, L.; Junod, P.; Litzistorf, G.; Monbaron, P.; et al. Long-term performance of the SwissQuantum quantum key distribution network in a field environment. *New J. Phys.* **2011**, *13*, 123001. [CrossRef]
17. Chen, T.Y.; Liang, H.; Liu, Y.; Cai, W.Q.; Ju, L.; Liu, W.Y.; Wang, J.; Yin, H.; Chen, K.; Chen, Z.B.; et al. Field test of a practical secure communication network with decoy-state quantum cryptography. *Opt. Express* **2009**, *17*, 6540–6549. [CrossRef] [PubMed]
18. Chen, T.Y.; Wang, J.; Liang, H.; Liu, W.Y.; Liu, Y.; Jiang, X.; Wang, Y.; Wan, X.; Cai, W.Q.; Ju, L.; et al. Metropolitan all-pass and inter-city quantum communication network. *Opt. Express* **2010**, *18*, 27217–27225. [CrossRef] [PubMed]
19. Wang, S.; Chen, W.; Yin, Z.Q.; Zhang, Y.; Zhang, T.; Li, H.W.; Xu, F.X.; Zhou, Z.; Yang, Y.; Huang, D.J.; et al. Field test of wavelength-saving quantum key distribution network. *Opt. Lett.* **2010**, *35*, 2454–2456. [CrossRef] [PubMed]
20. Sasaki, M.; Fujiwara, M.; Ishizuka, H.; Klaus, W.; Wakui, K.; Takeoka, M.; Miki, S.; Yamashita, T.; Wang, Z.; Tanaka, A.; et al. Field test of quantum key distribution in the Tokyo QKD Network. *Opt. Express* **2011**, *19*, 10387–10409. [CrossRef] [PubMed]
21. Alleaume, R.; Roueff, F.; Diamanti, E.; Lütkenhaus, N. Topological optimization of quantum key distribution networks. *New J. Phys.* **2009**, *11*, 075002. [CrossRef]
22. Tanizawa, Y.; Takahashi, R.; Dixon, A.R. A routing method designed for a Quantum Key Distribution network. In Proceedings of the 2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN), Vienna, Austria, 5–8 July 2016; pp. 208–214. [CrossRef]
23. Caleffi, M. Optimal Routing for Quantum Networks. *IEEE Access* **2017**, *5*, 22299–22312. [CrossRef]
24. Yang, C.; Zhang, H.; Su, J. The QKD network: Model and routing scheme. *J. Mod. Opt.* **2017**, *64*, 2350–2362. [CrossRef]
25. Ma, C.; Guo, Y.; Su, J. A multiple paths scheme with labels for key distribution on quantum key distribution network. In Proceedings of the 2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing China, 25–26 March 2017; pp. 2513–2517.
26. Tysowski, P.K.; Ling, X.; Lütkenhaus, N.; Mosca, M. The engineering of a scalable multi-site communications system utilizing quantum key distribution (QKD). *Quantum Sci. Technol.* **2018**, *3*, 024001. [CrossRef]
27. Mehic, M.; Niemiec, M.; Rass, S.; Ma, J.; Peev, M.; Aguado, A.; Martin, V.; Schauer, S.; Poppe, A.; Pacher, C.; et al. Quantum Key Distribution: A Networking Perspective. *ACM Comput. Surv.* **2020**, *53*, 96. [CrossRef]
28. Amer, O.; Krawec, W.O.; Wang, B. Efficient Routing for Quantum Key Distribution Networks. In Proceedings of the 2020 IEEE International Conference on Quantum Computing and Engineering (QCE), Denver, CO, USA, 12–16 October 2020; pp. 137–147. [CrossRef]
29. 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]
30. Chen, L.Q.; Chen, J.Q.; Chen, Q.Y.; Zhao, Y.L. A quantum key distribution routing scheme for hybrid-trusted QKD network system. *Quantum Inf. Process.* **2023**, *22*, 75. [CrossRef]
31. Bi, L.; Miao, M.; Di, X. A Dynamic-Routing Algorithm Based on a Virtual Quantum Key Distribution Network. *Appl. Sci.* **2023**, *13*, 8690. [CrossRef]
32. Dervisevic, E.; Tankovic, A.; Fazel, E.; Kompella, R.; Fazio, P.; Voznak, M.; Mehic, M. Quantum Key Distribution Networks—Key Management: A Survey. *arXiv* **2024**, arXiv:2408.04580.
33. Tang, X.; Ma, L.; Mink, A.; Nakassis, A.; Xu, H.; Hershman, B.; Bienfang, J.; Su, D.; Boisvert, R.F.; Clark, C.; et al. Demonstration of an active quantum key distribution network. In Proceedings of the Quantum Communications and Quantum Imaging IV, San Diego, CA, USA, 13–14 August 2006; Meyers, R.E., Shih, Y., Deacon, K.S., Eds.; International Society for Optics and Photonics (SPIE): Pamiers, France, 2006; Volume 6305, p. 630506. [CrossRef]
34. Tayduganov, A.; Rodimin, V.; Kiktenko, E.O.; Kurochkin, V.; Krivoshein, E.; Khanenkov, S.; Usova, V.; Stefanenko, L.; Kurochkin, Y.; Fedorov, A.K. Optimizing the deployment of quantum key distribution switch-based networks. *Opt. Express* **2021**, *29*, 24884–24898. [CrossRef] [PubMed]

35. Salvail, L.; Peev, M.; Diamanti, E.; Alléaume, R.; Lütkenhaus, N.; Länger, T. Security of trusted repeater quantum key distribution networks. *J. Comput. Secur.* **2010**, *18*, 61–87. [CrossRef]
36. Zhou, H.; Lv, K.; Huang, L.; Ma, X. Quantum network: Security assessment and key management. *IEEE/ACM Trans. Netw.* **2022**, *30*, 1328–1339. [CrossRef]
37. Gaidash, A.; Miroshnichenko, G.; Kozubov, A. Quantum network security dependent on the connection density between trusted nodes. *J. Opt. Commun. Netw.* **2022**, *14*, 934–943. [CrossRef]
38. Solomons, N.R.; Fletcher, A.I.; Aktas, D.; Venkatachalam, N.; Wengerowsky, S.; Lončarić, M.; Neumann, S.P.; Liu, B.; Samec, Ž.; Stipčević, M.; et al. Scalable authentication and optimal flooding in a quantum network. *PRX Quantum* **2022**, *3*, 020311. [CrossRef]
39. Stępniak, M.; Mielczarek, J. Analysis of multiple overlapping paths algorithms for secure key exchange in large-scale quantum networks. *J. Inf. Secur. Appl.* **2023**, *78*, 103581. [CrossRef]
40. Chen, T.Y.; Jiang, X.; Tang, S.B.; Zhou, L.; Yuan, X.; Zhou, H.; Wang, J.; Liu, Y.; Chen, L.K.; Liu, W.Y.; et al. Implementation of a 46-node quantum metropolitan area network. *npj Quantum Inf.* **2021**, *7*, 134. [CrossRef]
41. Kirsanov, N.S.; Pastushenko, V.A.; Kodukhov, A.D.; Yarovikov, M.V.; Sagingalieva, A.B.; Kronberg, D.A.; Pflitsch, M.; Vinokur, V.M. Forty thousand kilometers under quantum protection. *Sci. Rep.* **2023**, *13*, 8756. [CrossRef] [PubMed]
42. Kirsanov, N.; Pastushenko, V.; Kodukhov, A.; Aliev, A.; Yarovikov, M.; Strizhak, D.; Zarubin, I.; Smirnov, A.; Pflitsch, M.; Vinokur, V. Loss Control-Based Key Distribution under Quantum Protection. *Entropy* **2024**, *26*, 437. [CrossRef]
43. Wang, X.B. Beating the Photon-Number-Splitting Attack in Practical Quantum Cryptography. *Phys. Rev. Lett.* **2005**, *94*, 230503. [CrossRef] [PubMed]
44. Ma, X.; Qi, B.; Zhao, Y.; Lo, H.K. Practical decoy state for quantum key distribution. *Phys. Rev. A* **2005**, *72*, 012326. [CrossRef]
45. Zhang, Z.; Zhao, Q.; Razavi, M.; Ma, X. Improved key-rate bounds for practical decoy-state quantum-key-distribution systems. *Phys. Rev. A* **2017**, *95*, 012333. [CrossRef]
46. Trushechkin, A.S.; Kiktenko, E.O.; Fedorov, A.K. Practical issues in decoy-state quantum key distribution based on the central limit theorem. *Phys. Rev. A* **2017**, *96*, 022316. [CrossRef]
47. Trushechkin, A.S.; Kiktenko, E.O.; Kronberg, D.A.; Fedorov, A.K. Security of the decoy state method for quantum key distribution. *Physics-Uspekhi* **2021**, *64*, 88. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.