IEEE Access

Received 20 December 2022, accepted 19 February 2023, date of publication 2 March 2023, date of current version 7 March 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3251649

RESEARCH ARTICLE

A Novel Security Survival Model for Quantum Key Distribution Networks Enabled by Software-Defined Networking

OMAR SHIRKO, (Graduate Student Member, IEEE), AND SHAVAN ASKAR, (Senior Member, IEEE)

Department of Information System Engineering, Erbil Polytechnic University, Erbil 44001, Iraq

Corresponding author: Shavan Askar (shavan.askar@epu.edu.iq)

ABSTRACT Quantum key distribution (QKD) is a technique for distributing symmetric encryption keys securely using quantum physics. The rate of key distribution is low and decreases exponentially with increasing distance. A classic trusted relay (CTR) uses additional keys to enhance security distance in QKD networks. In practice, the assurance of security for certain relay nodes is still lacking, despite the fact that CTR requires that all nodes be trusted. Owing to channel unreliability, system faults accumulate during the key relay, thereby increasing the probability of CTR failing to distribute the secret key. The failure of a successful key relay would then result in the subsequent destruction of all the keys involved in the process, which leads to the wasting of the quantum secret key and reduction system encryption. Hence, alleviating the effect of CTR failure for the purpose of obtaining key security distribution of distant quantum network is necessary issue to tackle. Therefore, a new scheme is needed in order to overcome the above-mentioned issues to come up with a better utilization of the generated keys. In this study, a software-defined networking (SDN) technique is introduced to circumvent this drawback by utilising the flexibility provided by the SDN paradigm for better QKD network management. In particular, a novel survivability model called software-defined quantum key relay failure (SDQKRF) is proposed in this paper in which a new function is developed and added to the SDN controller. According to the simulation results, SDN over a QKD network using the SDQKRF model is more reliable and performs better in terms of the key generation ratio, key utilisation rate, recovery after failure, avalanche effect, and service blocking rate than a regular QKD network without the SDQTRF model.

INDEX TERMS Quantum key distribution (QKD), software-defined network (SDN), survivability, classical trusted relay (CTR).

I. INTRODUCTION

It is expected that by 2023, approximately two-thirds of the world population will have Internet access, this suggests that the amount of Internet users is estimated to will increase from 3.9 billion (51% of the world population) in 2018 to 5.3 billion (66% of the world population) in 2023 [1]. The increase in internet access will lead to an increase in the number of security breaches such as eavesdropping and data interception, which consequently can result in the loss of personal information, financial losses, and significant disruptions to services [2], [3]. Therefore, cryptographic techniques became an inevitable alternative to ensure the safety of communication carried out through the internet [4]. However, one of the most essential cryptographic tasks is to establish secure cryptographic keys across untrusted networks [5]. Traditionally, encryption methods based on public-key cryptography have been used, enabling cryptographic keys to be distributed over unreliable networks. Although public-key cryptography security relies on the computational complexity

The associate editor coordinating the review of this manuscript and approving it for publication was Abderrahmane Lakas.

of mathematical functions, the rapid growth of processor chips and quantum computers has rendered communication security far less reliable [25], [53], and [6]. Hence, the current encryption techniques are insufficient to guarantee security in the quantum-computing era. Therefore, a new approach is required to protect the data transported across communication networks from these security lapses [26]. Because a quantum bit is uncopiable, any attempts to do so will be detected easily by both the sender and receiver. Quantum key distribution (QKD) is one of the most promising alternatives to traditional data encryption methods [27]. QKD is derived from the fundamental principles of quantum mechanics, including the Heisenberg uncertainty principle and quantum no-cloning theorem [7], [8]. QKD can create a secure link between two distant parties using quantum secret keys [9], [10], [41]. However, QKD has mostly been used for point-to-point communications. Despite improvements in this direction, the performance of point-to-point QKD networks remains fundamentally constrained in terms of distance limitation and rate because secret key resources are typically limited in current advanced QKD systems [16], [10]. This issue can be resolved using a series of QKD relays, known as the classical trusted relay (CTR) [54]. Hence, the secret keys in QKD networks are valuable [10], where one of the main goals of the CTR technology is to send quantum keys to distant QKD nodes using highly secure encryption. In addition, the nodes of the CTR technique are expected to be safe from intrusion and attack by any unauthorised party [28]. In CTR technology, every relay node must be trusted; however, certain relay segments are unsecured [4]. Moreover, the operation of the key relay occurs via the public channel of the QKD system, this means that it would be impossible to get the signal immune to eavesdropping. This implies that, if one amongst the CTR nodes is compromised, the entire network is considered insecure [4]. Under such a condition, a compromised CTR node implies that the CTR technique failed to distribute quantum secret keys across QKD systems [22]. In this case, there is a higher demand for QKD network secret keys and it is difficult to meet the security needs of the service. This is one of the major challenges in QKD-network-based CTR technology. Therefore, it became a necessity to come up with a new survivability scheme for the CTR technique, moreover, the control and management of the quantum keys in the relay process needs improvement [6]. Despite the waste of resources and complexity incurred by CTR, a flexible and effective QKD network can be realised using software-defined networking (SDN). SDN enables the separation of control (management) and data (forwarding) planes [5]. SDN enables new technologies and services to be added more quickly and allows for centralisation of management and optimisation based on the principles of network programmability and configurability. Artificial intelligence (AI), machine learning (ML), deep learning (DL), and optimisation techniques can play significant roles in enhancing the performance of existing QKD networking techniques [6]. Consequently, there is a growing interest in employing machine learning to enhance the performance of quantum communication networks [36]. One of the highly used algorithms in this regard is the reinforcement learning (RL) algorithm which is used to explain and conclude how an intelligent agent learns and improves its strategies through interacting with its environment [55], [17], [15], and [49].

In this paper, we propose a novel survivability model called software-defined quantum trusted relay failure (SDQTRF) to overcome the challenges of the CTR technique failure based on the QKD network. In the proposed model, the SDN controller is responsible for alleviating the effect of CTR technology failure. This paper presents three main contributions, they are namely; (1) the introduction of a novel SDN controller in which a new function is added, in addition, a new relay protocol has been proposed to enhance the management of unsuccessful relayed keys; (2) a novel concept has been presented to improve the security of secret key recycling by adding an RL (Q-learning) algorithm to increase the survivability of quantum secret keys that were not successfully relayed; and (3) a new routing method for finding an alternative secure path has been presented, it has an effective role in case failing to relay the recycled secret keys.

The paper is structured as follows; the work conducted regarding QKD-based CTR technology and the motivation behind proposing SDQKRF are presented in Section II. Section III presents the basic concepts of QKD and the essential principles of CTR technology. The architecture of QKD over SDN is described in Section IV. Section V presents the system model and the notation used in this paper. Section VI describes the proposed SDQKRF model. The algorithm for the proposed model is described in Section VII. Section VIII presents the performance evaluation of the proposed SDQKRF model. Finally, Section IX concludes the paper.

II. RELATED WORK AND MOTIVATION

Three different relay-based solutions are available in QKD-protected optical networks for secure long-distance communication. First, QKD-based quantum repeaters (quantum repeaters create an entangled state between two nodes located in different locations using the quantum entanglement principle to establish secure long-distance communication). [56], [6]. Second, the measurement-device-independent quantum key distribution (MDI-QKD) protocol [11], [12], [13], [14] (Based upon two-photon interference, the MDI-QKD protocol, is immune to all attacks against the detection system, which allows a QKD network with untrusted relays, fibre-based implementations aimed at longer distances, higher key rates, and network verification have been rapidly developed.) [57]. Third, QKD-based CTR technology (the generated secret keys on the first QKD link are further encrypted with the generated secret keys in the intermediate nodes before being relayed to the final destination node) [10], [6]. As the MDI protocol has limited safety distance (still limited to ~500 km) [18], [58] and quantum

repeaters are still undergoing development [56], most QKD networks are still deploying the CTR technology to enable communication over longer distances [10]. However, nodes with CTR technology still ought to be credible since they recognise the secret keys between the source and destination nodes [24]. To improve the security of the CTR technique, a new hybrid trusted/untrusted node in a CTR-based QKD network architecture was proposed in [19], [20], [21]. In addition, in [52], a topological abstraction-based protection strategy was proposed for a QKD network with partially trusted CTR nodes and a key protection threshold to change how key resources are shared. Furthermore, a QKD technique for a ring network was developed in [35] under similar trusted/ untrusted CTR technology nodes, which solves the security issue of key distribution in a ring backbone network. To reduce the wastage of quantum keys, a new concept involving the recycling of quantum keys was proposed in [22], which focused on the processing of failed relay secret keys based on CTR technology. However, few quantum-key recycling mechanisms have been proposed and strategies for quantum key recycling and reuse are required to increase the number of available keys [6]. Quantum keys are important in QKD-secured optical networks because their secret key rate is low [10]. Therefore, to improve the management of secret keys in QKD-based CTR technology, SDN has the potential for management in QKD networks [33]. Quantum key pool (QKP) technology was developed to facilitate the volume allocation of on-demand secret keys for control and data channels in software-defined optical networks (SDON) secured by QKD-based CTR technology [23]. Moreover, a time-scheduled approach for QKP construction was developed in [24] to facilitate efficient scheduling of QKD based on CTR across classical networks. In [29], SDN was used to provide services to several tenants over a QKD metropolitan network efficiently and flexibly. It was proven in [30] that SDN-controlled QKD networks can be employed to provide end-to-end keys for demand service provisioning based on the CTR technology. A practical key management scheme was presented in [31] for a QKD network, in which the key relay through the CTR technology is dynamically routed by SDN. Recently, machine learning (ML) has been actively used to improve the performance of QKD networks. In [32], the authors proposed an ML model based on a hybrid quantum-classical QKD network enabled by SDN to evaluate the performance of the quantum channel in terms of noise, secret key rate (SKR), and the presence of a classical channel. For QKD-secured optical networks based on CTR technology, a multi-tenant secret key assignment strategy based on reinforcement learning (RL) was proposed in [34]. A comparative study of heuristics and an RL-based approach was conducted by [33] to investigate the effectiveness of multitenant provisioning over a QKD network based on the CTR technique.

According to the above literature review, almost all experiments used CTR technology to increase the coverage range of QKD systems. In fact, this strategy does not come without it is downsides. If the security of certain CTR nodes cannot be guaranteed, which may be due to the activity of an eavesdropper or malicious attack, or if any of the CTR nodes are hacked, this trusted relay technology will be inefficient for the remote distribution of quantum keys. In this scenario, the failed key distribution based on the CTR technique increases the influence of numerous network issues, including the security needs of communication across networks, increasing the network key demand, secret key rate, QKD service blocking rate, and transmission distance. Moreover, most studies in the literature that are mentioned in this section focus on successfully distributing quantum secret keys based on the CTR technique. Hence, only a few studies have addressed the failure of the CTR mechanisms. In this study, we addressed this issue by proposing a new survivability model (SDQTRF). As far as we can tell, no work has been conducted to utilise SDN over QKD networks to manage the secret key when the CTR technique fails to distribute quantum keys. The SDQTRF model, which is based on applying SDN to a QKD network, was designed to improve the performance of the QKD network by minimising the effects of CTR failure.

III. FUNDAMENTAL PRINCIPLES OF QKD

This section focuses on the critical support technologies for QKD networks. First, the classic point-to-point QKD technique employing the BB84 and SARG04 protocols was presented. Finally, a QKD long-distance transmission scheme featuring ease of implementation was introduced based on the objective of increasing the key rate.

A. MECHANISM OF POINT-TO-POINT QKD

The basic principle behind point-to-point QKD was introduced in the first QKD protocol, that is, the BB84 protocol that was proposed by Bennett and Brassard in 1984 [37], as illustrated in Figure 2, which allows Alice and Bob to generate, transfer, and synchronise keys [10]. A "QKD link" refers to the logical connection between two remote QKD nodes that share a quantum channel for photon transmissions and a public channel for the post-processing of the information transferred [5]. The following is a description of how the secret key generation process works when using the BB84 protocol [6].

Phase 1: The Quantum State Alice makes a string of qubits, and for each qubit and chooses at random either a rectilinear basis (+) with two polarisation of photons (90°, 0°) or a diagonal basis (×) with two polarisations of photons (135°, 45°) [6]. Subsequently, a string of qubits is sent to Bob via a quantum channel. By contrast, Bob chooses a random measurement basis for each qubit received. Using the chosen basis, if the measuring bases between Alice and Bob are the same, it gives a perfectly correlated result, and Bob records a string of all received qubits, called the raw key. If the measurement bases differed, an uncorrelated result was obtained.

Phase 2: Alice and Bob communicate with each other over a public channel during post-processing to derive secret keys

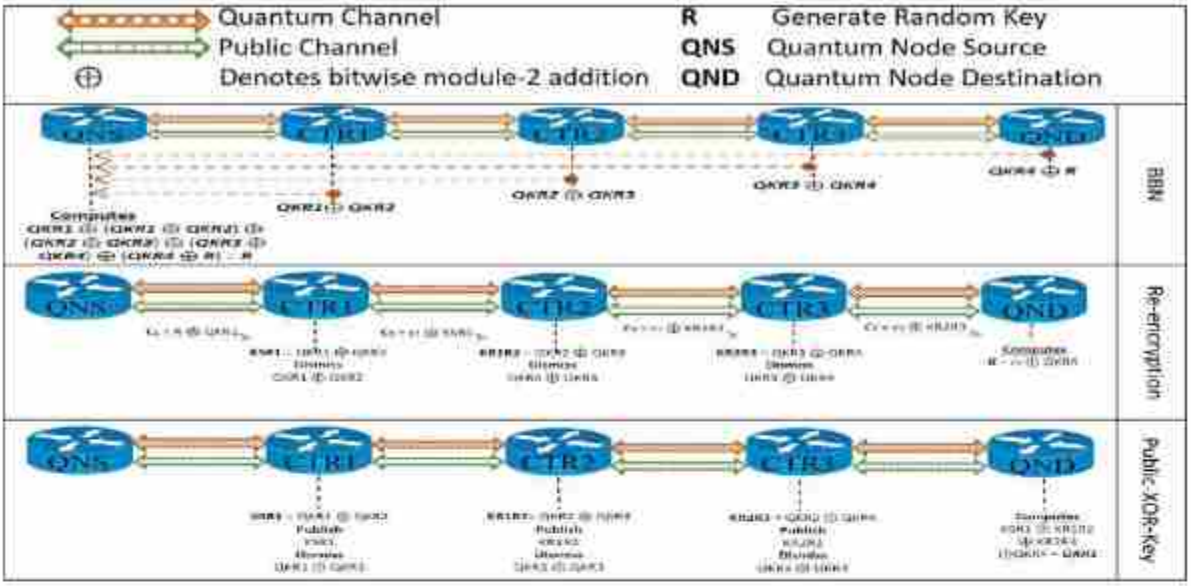

FIGURE 1. Examples of the models of BBN, Re-encryption and Public –XOR-Key schemes with two endpoints and three CTR nodes.

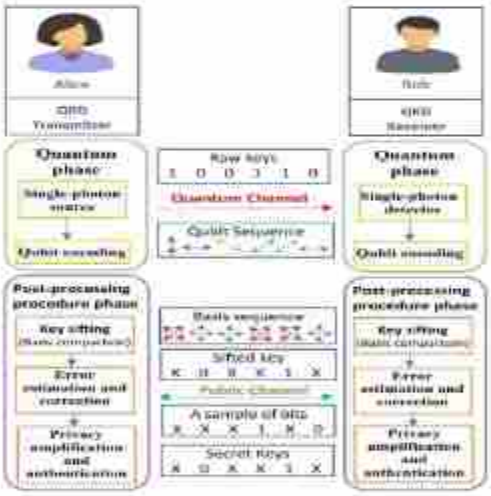

FIGURE 2. Point-to-point QKD mechanism based on BB84 protocol.

from the measurement results. The post-processing technique requires the following steps to obtain the secret key:

1) Sifting: Alice and Bob use a classical channel to exchange information about transmitted and received photons. Qubits belonging to the same measuring bases selected by Alice and Bob were retained; however, those corresponding to different measuring bases were eliminated, and the length of the key after the sifting the sifting stage = 1/2 of the raw key. Sifted key is generated by decoding the remaining qubits into a string of classical bits [10].
1) Error estimation and correction: The purpose of this step was to eliminate any risk of mistakes that may have occurred during the process of sifting. Alice and Bob used a public channel to communicate with one another and compare the results of exchanging a random substring of classical bits in sifted keys [10].
2) Privacy amplification and authentication: This step minimises the information of the secret key against a minimal number of unauthenticated users, and generates a new shorter key through the use of universal hash functions. In addition, an authentication procedure is necessary to ensure the safety of the secret generated by eavesdropping [6].

The SARG04 protocol was first proposed by Scarani et al. [38] in 2004. When the mechanism of point-to-point QKD of the BB84 protocol was compared to that of the SARG04 protocol, it was discovered that the first phases of both were the same [59]. The traditional post-processing procedure, which is the main difference between the two protocols, makes the SARG04 protocol more secure. Alice does not explicitly declare her bases during the second phase when Alice and Bob decide which bits their bases match. Instead, Alice announces a pair of non-orthogonal states, one of which she uses to encode bits. Bob does the same thing. If Bob uses the correct basis, he measures the correct state. If Bob makes the wrong choice, he will not be able to measure any of Alice's states, which will prevent him from identifying the bit. In the case of no error present, the length of the key remaining after the sifting step is 1/4 that of the raw key [39].

B. CTR TECHNOLOGY FOR LONG-DISTANCE TRANSMISSION

The consequences of signal loss and decoherence inherent to most transport media, such as optical fibres, impede

long-distance communication [60]. Early implementations of QKD systems concentrated their attention primarily focused on the communication that occurred between the two endpoints [6]. Until recently, QKD has mostly been used in point-to-point communication scenarios. Although there had been progress made in this area, the performance of point-to-point QKD networks remains limited by distance and throughput [40]. This constraint can be circumvented using CTR technology. Using trusted relays based on the mechanism of CTR in QKD networks was first suggested in 2002 [4]. This allowed both ends of the contact to use a series of reliable relays to extend the distance. CTR protocols provide the basis for quantum secret keys sent over CTRs. Since 2003 [42], the BBN key relay protocol was reliably implemented in quantum networks. However, the BBN protocol has two major drawbacks: it is time-consuming to establish the key material and it demands complete trust among all communicating nodes. Schartner and Rass presented a strategy for re-encrypting the key distribution for key relays based on QKD systems [43]. However, in the re-encryption method, the relay nodes are still required to retain the XOR values of the QKD keys secret, and successive communication between adjoining relay nodes is still necessary [4]. This is because the XOR value was used to decrypt the QKD keys. The public-XOR-key method, which has been used for quantum-secure communication since 2013, involves nodes of CTRs making their XOR keys public [4]. This allows the endpoints to share a key. When compared with the re-encryption strategy, this scheme reduces the complexity of the system and facilitates the traffic of the relay nodes. However, it should be noted that these three types of CTR protocols are used to distribute the quantum secret key via a public channel. Figure 1. provides a description of the primary process of the BBN, as well as re-encryption and public XOR key protocols [4].

IV. QKD ENABLED BY SDN ARCHITECTURE

A more advanced method of adding QKD to transport networks is to use the most recent advances in networking technology, particularly in network management [45], [46]. In the QKD network management perspective, SDN can significantly improve the management effectiveness of QKD networks [5], [44]. SDN can provide QKD networks with efficient and straightforward management because of its programmable and adaptable centralised control mechanism [33]. The architecture of SDN-enabled QKD networks is comprised of three layers, as shown in Figure 3. These layers are the application, control, and infrastructure layer [48].

1) Application layer: This layer is located at the top of the SDN-enabled QKD network. It not only responds immediately to user requirements but also makes network resources easier for users to find. Therefore, it might be able to provide what consumers need in terms of topological visualisation and quality of service. In addition, the controller can enable the abstraction of network resources, such as light-path creation for QKD and routes for the generation of the secret key in the infrastructure layer through northbound interfaces. Both processes occur in the context of QKD [47].

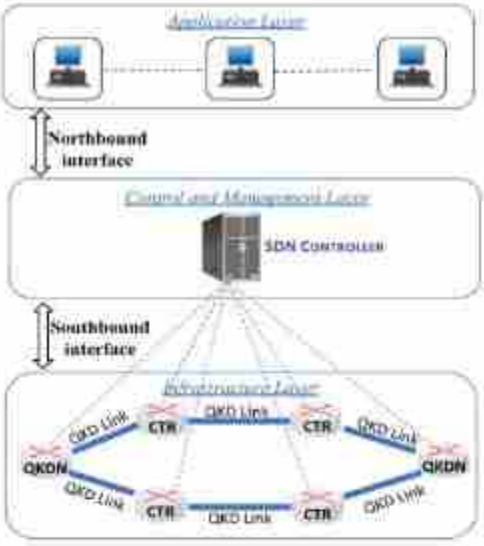

FIGURE 3. QKD over SDN architecture.

2) Control/management layer: This layer provides the operator with a comprehensive view of the QKD networks. This layer may have one or more controllers to manage the network in the infrastructure layer and make the network open to different applications. After receiving demands from the operators, the application layer creates requests that are subsequently transmitted to the controller via the northbound interface of the application layer. The controller then calculates and assigns QKD resources using a global network map and southbound interface protocol [10]. Correspondingly, the control layer manages QKD resources located in the infrastructure layer. Additionally, it is responsible for delivering services to multiple applications located in the application layer and receiving information regarding resource allocation and policy from the infrastructure layer [47].
3) Infrastructure layer: This layer lies at the bottom of the architecture concerning the performance of the QKD devices [47]. In the infrastructure layer, QKD nodes, also known as QKDNs, are connected to one another through QKD links. For long-distance end-to-end QKD, many CTR technology nodes are placed between two distant QKDNs [48]. The SDN controller and QKDN/CTR can communicate with each other and exchange messages through a southbound interface. Each QKDN/CTR can operate following the instructions received from the SDN controller over the southbound interface. There is a continuous production of a number of secret keys between any two QKDNs or CTRs that are directly linked to one another, as well as between a QKDN and CTR.

TABLE 1. Notation and definitions.

NOTATIONS	DEFINITIONS
QN	Q-value between pair nodes
SPn	new generate secure path
ACKn(i,j)	notification at success sending delivered
ACKfn(i,j)	notification at failure sending
Gkn	key generate for pair nodes
Rkn	recycling for pair nodes based on Q-values
Nfi	pair nodes failure
Nxi	next node in secure path , which mean CTR nodes
Nsf	nodes from source to failure node
ACKr	notification from controller to nsf to start recycling
S	Secure node
US	Unsecure nodes

V. SYSTEM MODEL

The proposed system model is described as follows: Graph G (N, C, SP, QKDn, SN, DN) represents a network that can

transmit secret keys. N is the set of secure path nodes; C is the connecting link, where every two nodes i and j are connected by a link (i, j), and all (i, j) ϵ C; SP represents the current secure path; QKDn represents the pairing key between pair nodes; SN represents the source node and DN represents the destination node. We assume that QKD can retain and manage end-to-end keys, considering that it can only produce point-to-point keys at a given rate through

direct links (i, j). Table 1 provides the list of notations and definitions that were used in this study.

VI. PROPOSED MODEL OF SDQKRF

In this section, we explain the suggested SDQTRF model for the purpose of alleviating the effect of CTR failure on QKD networks.

Three main objectives were behind coming up with SDQTRF paradigm. *First, how can the manageability of the QKD network be improved in the case of distributed the secret key fails?* For better control and management of the QKD network, we used SDN over the QKD network to handle the case of a failed secret key relay. However, the SDQTRF model does not affect the SDN process with QKD if the keys are successfully relayed. This indicates that the SDQTRF model will only take action when CTR technology fails to relay the secret keys. From this point, we designed a particular function, the *Contingency function*, within the SDN controller platform to guarantee that the SDQTRF model framework is more reliable and does not impact the SDN process with QKD in the case of a successful relay of the key. This function was regarded as the core of the SDQKRF framework. The *contingency function* is composed of a *Q-Learning module* and *Topology module*, which are both support modules.

Additionally, the key management procedure of the QKD network is significantly assisted by the relay key protocol, which significantly contributes to this process. In general, the suitable protocol is the public-XOR-key protocol, but in the case of CTR failure, the destination node will check if it is a bad key or an accepted key, which means that if it is a bad key, all the keys that have been relayed before will be destroyed. Thus, we proposed a new relay protocol that is suitable for unsuccessfully relayed keys. Moreover, the mechanism of the proposed relay protocol is centred upon the concepts of the public XOR-key protocol with some improvements. The following are a detailed explanation of the proposed relay protocol.

1) The QKD protocol is executed by each node along with its neighbouring nodes to produce *n* pairs of QKD keys, as illustrated in Figure 4.
2) The first relay node conducts the XOR operation using the QKD keys and adds the checksum of the XOR key. The relay node then temporarily stores the QKD keys and transmits the XOR key and its checksum to the subsequent relay node.
3) The next relay node performs the XOR key checksum. If no errors were observed, this process will be repeated. Then, all XOR results are sent to the next CTR node.
4) The destination node calculates the final key using the QKD key (QKN). The common key, Qk1, is then shared by the sender and the receiver. Subsequently, Alice uses key Qk1 to send a secret message to Bob.

Remark1: When the secret key is successfully relayed, the receiver asks all previous nodes to immediately dismiss the QKD keys.

Remark2: The following paragraphs explain the mechanism of the relay protocol in the case of CTR failure.

Second, how can unsuccessful relay keys be utilised to expand key availability? The most efficient way to collect and process keys that cannot be relayed is to recycle the failed keys and use them to encrypt the services again but with a lower level of security. However, the reliability of the recycling process could be improved by determining the recycling amount. To make the recycling process more secure, we used the recycling method in SDQTRF, but with certain improvements. With reference to the *Q-value*, the *Q-Learning module* of the SDQTRF model determines the recycling amount. In addition, we ensured that the key recycling process was only carried out from the source node to the nodes suspected to have been the source of failure in the key distribution process. Further clarification is provided in the following example.

- We assumed the secure path contains five nodes.
- The secret key is sent from the source node (1) to the destination node (5).
- We assumed that the failure occurred between nodes (3 and 4).

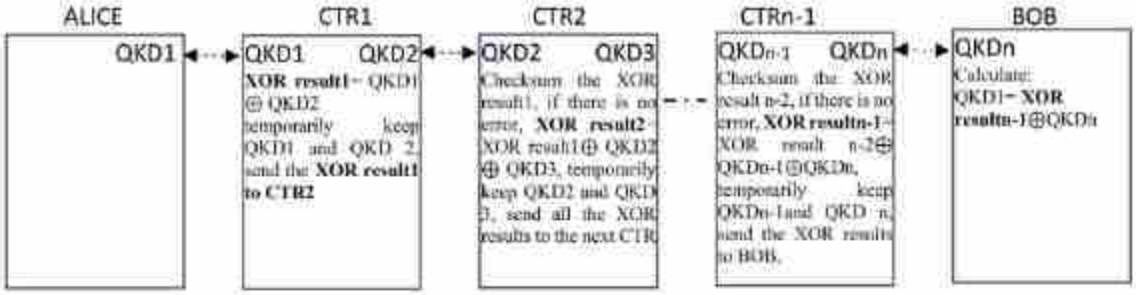

FIGURE 4. Proposed relay key protocol.

FIGURE 5. Contingency function inside the controller.

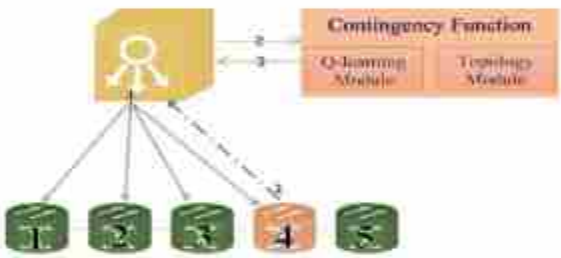

FIGURE 6. Example of the recycling process.

1) Node 4 sends a notification to the controller, as demonstrated in Figure 6.
2) The controller will then request the operation of the contingency function.

If the contingency function determines that there are no two successive failures in the same pair of nodes, then the Q-learning module computes the Q-values for the secure path nodes (1–4) and sends them to the controller. The contingency function stores information on failures between pairs of nodes (3 and 4) in the topology module.

3) The controller then sends the Q-values to nodes (1–4) and asks the nodes (1–4) to recycle their keys based on the Q-values. Then, it sends a key from the source node to the destination node.

Remark3: The topology module is used to collect, store, and update the QKD network topology as well as the information of the CTR nodes, which will take it periodically from the controller.

Third, what is the alternative plan if the proposed model fails to reuse the failed relay keys, which was based on the recycling method? One method considered as a potential solution to this issue is to find a different route that can be guaranteed to be safe. From this perspective, we propose a new idea to find a new secure path and make it suitable for the SDQTRF framework. Before finding the secure path, based on the occurrence of two successive failures at the same node, we assume that there are secure and unsecure nodes (note that the unsecure nodes are assumed to be the cause of the keys' failure to relay successfully). An appropriate secure path in the SDQTRF model is identified using a Q-learning module. In accordance with the Q-learning module, we allocated two environments: a secure environment with secure nodes and an insecure environment with insecure nodes. Using the topology module, secure and unsecured nodes were saved for the current and previous topologies. Further clarification is provided in the following example.

- In the last example, we assume that there are two successive failures in the same pair of nodes (3 and 4), as shown in Figure 7.

1) Retransmit notification from node 4 to the controller.
2) The controller will then request the operation of the contingency function.
3) If the contingency function determines, based on previously saved information in the topology module, that there are two consecutive failures at the same pair of nodes, then it will mark nodes (3 and 4) as unsecure nodes and initiate the Q-learning module to find a new secure path excluding the unsecure nodes (3 and 4) and send it to the controller.
4) The controller then asks the nodes in the new secure path to generate a new secret key between each pair of nodes and sends the key from the source node to the destination node.

VII. ALGORITHM IMPLEMENTATION OF SDQTRF MODEL

The proposed SDQTRF model in this study provides an optimal solution for minimising the impact of CTR failure. The overall steps of the SDQTRF mechanism in accordance with the notations provided in Table 1 are listed in Table 2. Lines 1 and 2 were used for initialisation. The for-loop covering lines 3–28 starts sending QKD keys from the source

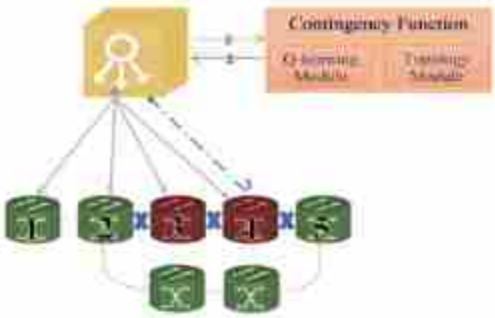

FIGURE 7. The proposed process to find a new secure path.

node to the destination node. Line 4 sends the key to the next node (Nx_i), whereas line 5 applies the relay protocol to verify whether the next node has successfully received the key. If it is successful, line 7 checks whether the key has reached the destination node. If it reaches the destination node, line 8 goes to the output of the algorithm. If not, line 10 returns to send the key to the next node (line 3). If the check of acknowledgement in line 6 were a failure then in line 13 the received node will send a failure acknowledgement to the controller to inform it that there was error occurred, in line 14 the controller will run the contingency function, in line 15 the contingency function requests that the topology module collect information regarding each node. In line 16, the contingency function is verified. then in line 17 the contingency function will mark the two sender and receiver nodes as unsecured nodes, then in line 18 it will update the topology module and exclude these unsecured notes from the new secure path, in line 19 new secure path will be generated based on the exclusion that was determined in line 17 & 18 after new secure path was found the algorithm will start all over again from line 3, else if the line 16 were false (no successive failure are found) then in line 22 the contingency function will ask the Q-learning module to generate Q-value for each secure path nodes and will ask the controller to send notification from source node to the node was error occur to start key recycling depending on the generated Q-values as in lines 23 & 24, then after key recycling is done the algorithm will start all over again from line 3, and continue till key reach to the destination node.

As shown in Table 2, the SDQTRF model is described. A secure path can be determined according to SDN. As a result, the selected path has fewer total hops. Then, according to the analysis of the CTR node, the key from Sn to Dn will begin to be sent. Subsequently, the model then further checks if it has delivered the key successfully hop by hop based on the proposed relay protocol, If untrusted nodes are absent in the path, the key will reach the destination node successfully. If there is an untrusted node and a failure is detected by a CTR node, an ACK_{fo} will be sent to the controller, which will run the contingency function. The contingency function will check if two successive failures occur at the same node.

TABLE 2. SDQTRF algorithm.

	Algorithm: The proposed SDQTRF model
1:	mputs G (N, C, SP, QKDn, SN, DN)
2:	output (Success Delivered Key from SN to DN then dismiss the QKD key)
3:	For i=SN to DN Loop
4:	Send Key to Nx_i
5:	Based on the proposed relay protocol, check if delivery was successful.
6:	If ACKn(Nxi) then:
7:	If Nxi = DN then:
8:	output
9:	Else:
10:	Go to Step: 3
11:	End if
12:	Else :
13:	Send ACKfn(Nxi) to Controller.
14:	Controller run contingency function
15:	Contingency request network topology
16:	If twice successive failures at the same node then:
17:	Mark Nfi as uS and the rest nodes as S
18:	Excluding Nfi
19:	Generate SPn and Gkn
20:	Go to Step: 3
21:	Else:
22:	Generate QN value for Nxf
23:	Controller sends ACKr
24:	Start Rkn (Nxf)
25:	Go to Step: 3
26:	End if
27:	End if
28:	End Loop

If no, then it will generate Q-Values from Sn to the node where the failure occurred, and the controller will send these values to nodes and perform a key recycling, then resume sending the key. If two successive failures occur, then the contingency function will mark the pair of nodes as unsecure and start Q-learning to generate a new secure path excluding

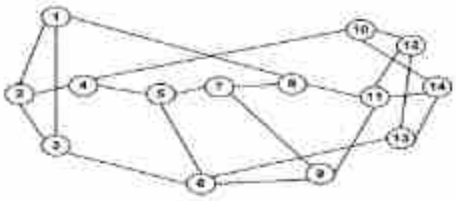

FIGURE 8. NSFNET network topolog.

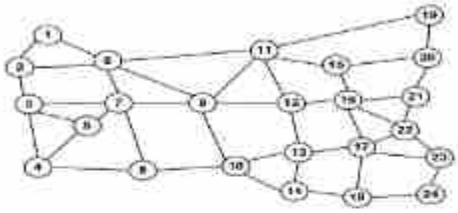

FIGURE 9. USNET network topology.

the unsecure nodes. Once the new secure path is made, the controller will start sending the key one hop at a time again.

VIII. SIMULATION RESULTS

In order to verify and evaluate the performance of the SDQTRF model, two types of network topology were used in the simulation: The National Science Foundation Network (NSFNET, 14 nodes and 21 links) and the United States network (USNET, 24 nodes and 43 links), as shown in Figure 8 and 9, respectively.

Python was utilized as a programming language for the purpose of simulating the proposed SDQTRF model. The hardware environment consists of single GPU NVIDIA GeForce RTX 3060Ti, Windows 11 was utilized as an operation system on the workstation with CUDA 11.3. For our study, we focused on a scenario in which the key was unsuccessfully relayed. We assumed that SDN over the QKD network operated normally. In the simulation, we used an in-built Linear Congruential Random Number Generator (RNG) to generate random errors within the nodes to evaluate performance. The results are based on an average of 1000 simulations for both network topologies. The SARG04 Protocol was implemented for each pair of nodes to produce QKD keys, which were then utilised throughout the simulation. Different key lengths are used in the simulations. The simulation performance of the SDQTRF model is presented in table 3.

The main objective of the SDQTRF model was to reduce the impact of CTR failure on the QKD network. From this point on, we have evaluated and compared (with and without the SDQTRF model) the ratios of key generation, recovery after failure, avalanche-effect-total-failure, and service blocking rate.

A. KEY-GENERATION RATIO

The key-generation ratio, also known as the KGR, is considered to be one of the most significant ways to measure the

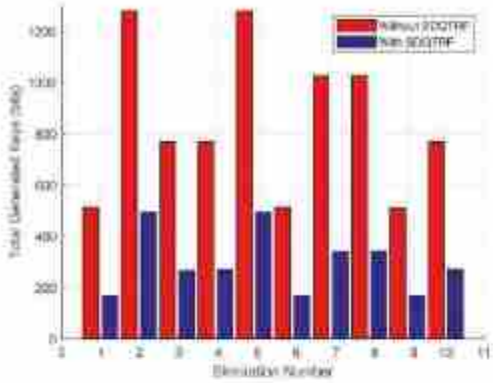

FIGURE 10. Simulation results: KGR of single failure after 10 simulated runs in NSFNET.

impact of CTR failure on a QKD network. However, the KGR was computed in the simulation at a rate of one failure per iteration. During the simulation, we examined how the KGR would work if there was one failure in a number of different situations that fell within a range of ten simulations. The KRG that occurs for every failure is shown in Equations (1) and (2), and is represented as follows:

$$\text{withoutSDQTRFmodel} = \text{ni} \times \text{kl} \tag{1}$$

$$\text{withSDQTRFmodel} = \sum_{i=\text{ni}}^{\text{ni}} Q_i \times \text{kl} \tag{2}$$

where ni, kl, and Q_i denote the index of the failed node in the topology, the key length, and the Q value, respectively.

In Figures 10 and 11, we computed the KGR for various scenarios, where each bar in the Figures represents a separate scenario in which there was a single failure. However, there was no connection between the scenarios. It can be observed in Figures 10 and 11 that the system that operates without SDQTRF generates secret keys more frequently than the system that operates with SDQTRF during the simulation run. Compared to the system operating without the SDQTRF model, the system operating with the SDQTRF model decreased the impact of CTR failure in terms of KGR in NSFNET and USNET (approximately 33% and 39%, respectively).

B. KEY UTILIZATION RATE

The key utilisation rate (KUR) was set up to measure how well the QKD network works if CTR fails. The KUR shows how much of the key was used out of the total key amount, and is expressed as follows [22]:

$$\text{kur} = \frac{N_{ku} + N_{kl}}{N_{kg}} \tag{3}$$

where N_{ku} the indicates number of keys the service has successfully utilised, N_{kl} represents the key stock in each CTR node at the end of the simulation, and N_{kg} shows how many keys were generated during the entire simulation run cycle.

TABLE 3. Average Simulation result for the mechanism of the SDQTRF model in the NSFNET and USNET topologies over the whole simulation run.

Simulation Run times till two successive failure occurs	NUMBER OF SUCCESSES SENDING	Number of Failure	Topology Type	Average Time	Key Length
336	275	59	USNET	$2.17 * 10^{-3}$ µs	128
306	296	8	NSFNET	$8.4 * 10^{-5}$ µs	128
496	398	96	USNET	$7.4 * 10^{-3}$ µs	192
400	371	27	NSFNET	$6.3 * 10^{-3}$ µs	192
695	560	133	USNET	$5.4 * 10^{-4}$ µs	256
238	217	19	NSFNET	$2.6 * 10^{-4}$ µs	256

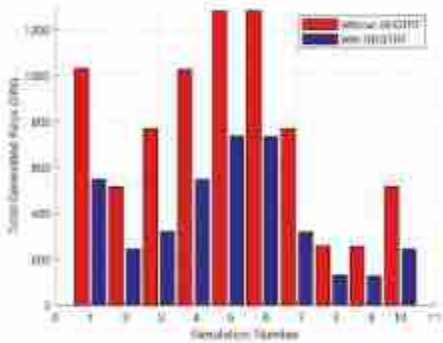

FIGURE 11. Simulation results: KGR of single failure after 10 simulated runs in USNET.

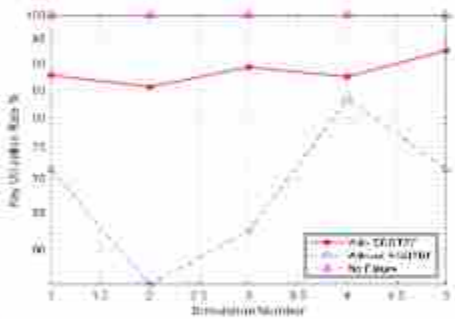

FIGURE 12. Simulation results: KUR for three cases (no failure, with and without SDQTRF model) based on one secure path with 5 nodes in NSFNET.

The trend of the KUR remained unchanged at 100% in the case of no failure, as shown in Figures 12 and 11, for all simulation runs. Similarly, as seen in Figures 12 and 11, without SDQTRF, the KUR decreases by almost 70%. The average KUR with SDQTRF was approximately 89% in NSFNET, which showed only a slight increase from approximately 3% to 92% in USNET.

C. RECOVERY AFTER FAILURE

Recovery after failure (RAF) is the network's performance after a failure occurs, as well as its action at failure and how long it takes to re-start sending after failure.

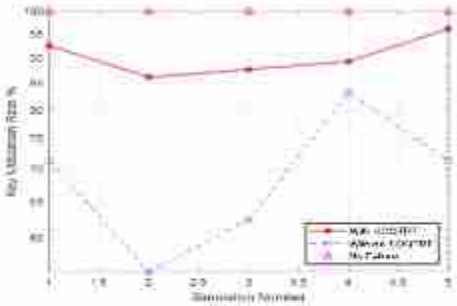

FIGURE 13. Simulation results: KUR for three cases (no failure, with and without SDQTRF model) based on one secure path with 5 nodes in USNET.

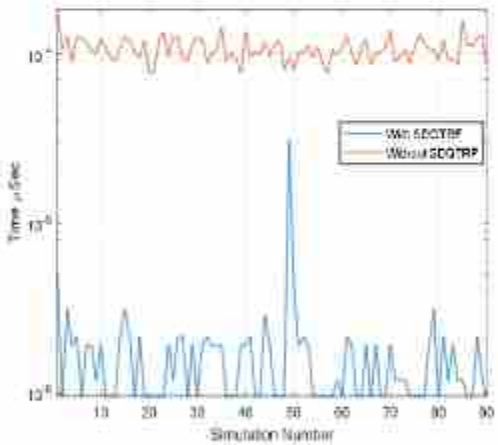

FIGURE 14. Simulation results: Time elapsed of RAF after 90 simulation runs in NSFNET.

Overall, the RAF process without SDQTRF takes much more time (in μs) than that with SDQTRF, as shown in Figures 14 and 15, which indicates that when using SDQTRF, the network can recover and re-start sending faster than without SDQTRF. It can be observed from Figures 14 and 15 that the average time of RAF in systems without SDQTRF in NSFNET is approximately $1*10^{-4}$ μs while in USNET, it is just above $1*10^{-4}$. In systems that work with SDQTRF, the average times of RAF in NSFNET and USNET are about ($1.5*10^{-6}$ μs and $2*10^{-6}$ μs respectively).

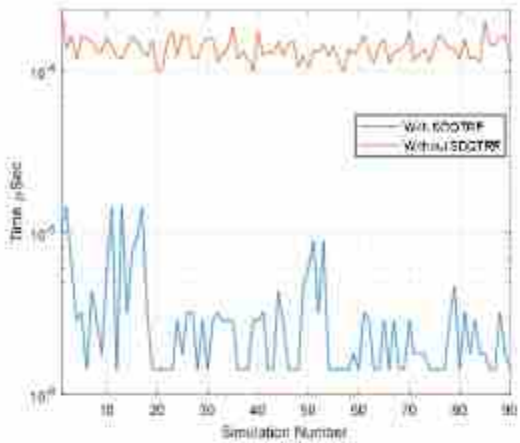

FIGURE 15. Simulation results: Time elapsed of RAF in 90 simulated runs in USNET.

D. AVALANCHE-EFFECT-TOTAL-FAILURE

The avalanche effect, also known as AETF, is required for all cryptographic algorithms. This effect causes progressively more important changes as information spreads through the structure of the algorithm. A piece or bit of the original secret key causes a significant change in an encrypted message. In our study, we employed the AETF to compare the avalanche quantity of the secret key with and without the SDQTRF model in the event of a CTR failure. However, rather than solving the avalanche effect, SQTRF was used to minimize the effect of avalanches on the key after failures is occurred. Without using the SQTRF model, the key is destroyed and a new key is generated, so the avalanche effect is high. This is expressed by the following equation [50].

$$\mathrm{AETF} = \left(\frac{\Sigma_i \mathrm{bitchange}}{\Sigma_i \mathrm{bittotal}}\right) \times 100 \tag{4}$$

Throughout the simulation runs, the avalanche of the secret key was seen to increase, regardless of whether the runs were with and without SDQTRF as shown in Figures 16 and 17. However, it was observed that the average number of key avalanches in a system without the SDQTRF model was almost double that of a system operating with the SDQTRF model because the key is destroyed and a new key is generated. In NSFNET and USNET, the trend of the average key avalanche began at 10% and peaked at 100% at the end of the run simulation. Approximately 5% of the key was avalanched in the first-run simulation of the system that works with the SDQTRF model in NSFNET, which then reached approximately 45% at the end of the run simulation, whereas in the USNET, it started at approximately 7% and ended at 55%.

E. SERVICE-BLOCKING RATE

For the performance evaluation, a performance criterion known as the success probability of QKD service requests was utilised, where the criterion is defined as the ratio of total accepted QKD service requests to total incoming QKD service requests. In addition, the success probability can be

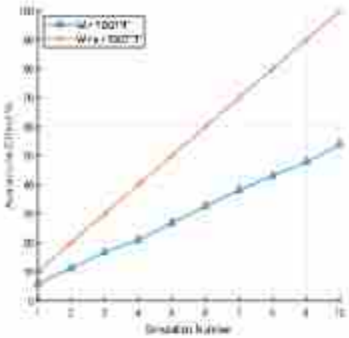

FIGURE 16. Simulation results: the trend of 10 simulation runs of the effectiveness of key avalanches, which were tested over the NSFNET.

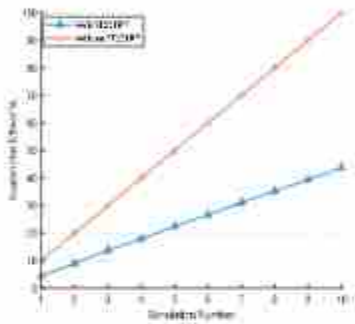

FIGURE 17. Simulation results: the trend of 10 simulation runs of the effectiveness of key avalanches, which were tested over the USNET.

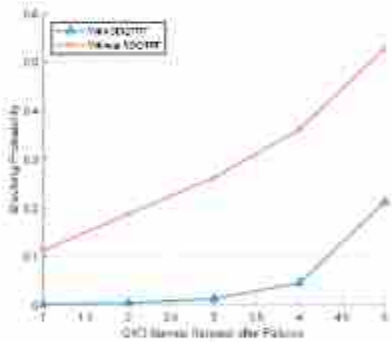

FIGURE 18. Simulation results: The service-blocking rate (SBR) when failures occur and how the system is requesting QKD service to re-generating secret keys after failure in NSFNET.

determined by the blocking probability because the sum of the success and blocking probabilities is equal to one [51]. A request for a QKD service may be denied or blocked for one of two reasons: failure in secret key rate assignment, or failure in secret key rate reassignment. Both failures can be detected when creating or modifying QKD services. Additionally, our simulation was based on a loss system (i.e., the Engest system [51]), which does not normally imply queuing. In this loss system, the value of the traffic load can be determined by examining the arrival and departure rates of QKD service requests, the traffic load represents the ratio of the arrival rate to the departure rate of QKD service requests.

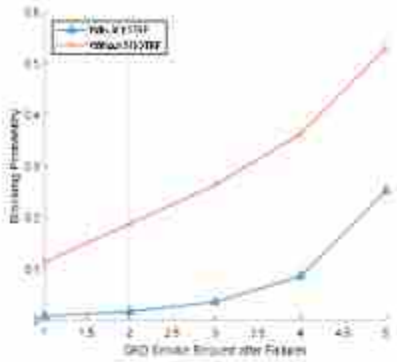

FIGURE 19. Simulation results: The service-blocking rate (SBR) when failures occur and how the system is requesting QKD service to re-generating secret keys after failure in USNET.

Comparing the curves in Figures 18 and 19, without utilising SDQTRF, the SBR is significantly greater than when SDQTRF is not used. In general, the SBR is not very high after one failure, and the system works the same with or without the SDQTRF. This is because the network can quickly fix the problem after a single failure. However, the chance of a service request being blocked increases after more failures. In Figures 18 and 19, the systems that work without the SDQTRF model start with just above 0.1 and end with just above 0.5, whereas in NSFNET, the trend of SBR in the system working with the SDQTRF model remained unchanged in the first two failures with a value of zero and then reached just above 0.2 at the final failure, compared to USNET, which started with 0 and then reached approximately 0.26 at the end.

IX. CONCLUSION AND FUTURE WORK

Currently, the CTR technology is the preferred practical solution for sending quantum information over long distances. However, if the security of certain CTR nodes unable to be guaranteed in practical systems, the CTR technique can be regarded as unreliable for the remote distribution of quantum keys. However, to minimise the impact of CTR failure on the QKD network, an efficient survivability model called SDQTRF was proposed in this study. With the help of the proposed new relay protocol, a new function has been proposed called the "Contingency Function" inside the SDN controller to improve key management in the case of an unsuccessful relay key. The quantity of recycling is determined using Q-learning for security improvements of the recycling process. To increase the survivability of the QKD network, a novel concept for finding a new secure path based on the Q-learning method was developed. In terms of KGR, KUR, RAF, AETF, and SBR, the performance of the proposed model was compared with and without utilisation of the SDQTRF model. To examine the effectiveness of the proposed SDQTRF, simulations were performed on two networks, NSFNET and USNET. Regardless of the values of KGR, KUR, RAF, AETF, and SBR, the simulation results indicate that the proposed SDQTRF model is superior to the system without the SDQTRF model. The SDQTRF model could be improved in the future by considering certain aspects that should be taken into account. Instead of excluding some CTR nodes when searching for a new secure path, the SDQTRF model can be used to overcome or assess the major reasons for CTR technology failure. In addition, owing to the high resource requirements of the Q-table, Q-learning cannot be utilised directly to enhance network routing. Therefore, the deep Q-learning approach can be used to enhance the SDQTRF model instead of the Q-learning method. The reason for this is that deep Q-learning employs neural networks to calculate Q-values instead of regular Q-tables, leading to more precise results.

REFERENCES

[1] Cisco. (Mar. 9, 2020). *Cisco Annual Internet Report—Cisco Annual Internet Report (2018–2023) White Paper*. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html

[2] N. Skorin-Kapov, M. Furdek, S. Zsigmond, and L. Wosinska, "Physical-layer security in evolving optical networks," *IEEE Commun. Mag.*, vol. 54, no. 8, pp. 110–117, Aug. 2016, doi: 10.1109/MCOM.2016.7537185.

[3] M. Furdek, N. Skorin-Kapov, S. Zsigmond, and L. Wosinska, "Vulnerabilities and security issues in optical networks," in *Proc. 16th Int. Conf. Transparent Opt. Netw. (ICTON)*, Jul. 2014, pp. 1–4, doi: 10.1109/ICTON.2014.6876451.

[4] H. Dong, Y. Song, and L. Yang, "Wide area key distribution network based on a quantum key distribution system," *Appl. Sci.*, vol. 9, no. 6, p. 1073, Mar. 2019, doi: 10.3390/app9061073.

[5] M. Mehic, M. Niemiec, S. Rass, J. Ma, M. Peev, A. Aguado, V. Martin, S. Schauer, A. Poppe, C. Pacher, and M. Voznak, "Quantum key distribution: A networking perspective," *ACM Comput. Surv.*, vol. 53, pp. 1–41, Sep. 2020.

[6] P. Sharma, A. Agrawal, V. Bhatia, S. Prakash, and A. K. Mishra, "Quantum key distribution secured optical networks: A survey," *IEEE Open J. Commun. Soc.*, vol. 2, pp. 2049–2083, 2021, doi: 10.1109/OJCOMS.2021.3106659.

[7] H.-K. Lo, M. Curty, and K. Tamaki, "Secure quantum key distribution," *Nature Photon.*, vol. 8, no. 8, pp. 595–604, Aug. 2014.

[8] V. Scarani, H. Bechmann-Pasquinucci, N. J. Cerf, M. Dušek, N. Lütkenhaus, and M. Peev, "The security of practical quantum key distribution," *Rev. Mod. Phys.*, vol. 81, no. 3, pp. 1301–1350, Sep. 2009.

[9] Y. Zhao, Y. Cao, W. Wang, H. Wang, X. Yu, J. Zhang, M. Tornatore, Y. Wu, and B. Mukherjee, "Resource allocation in optical networks secured by quantum key distribution," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 130–137, Aug. 2018.

[10] Y. Zhao, Y. Cao, X. Yu, and J. Zhang, "Quantum key distribution (QKD) over software-defined optical networks," in *Quantum Cryptography in Advanced Networks*, O. G. Morozov, Ed. Rijeka, Croatia: IntechOpen, 2019, ch. 2, doi: 10.5772/intechopen.80450.

[11] H.-K. Lo, M. Curty, and B. Qi, "Measurement-device-independent quantum key distribution," *Phys. Rev. Lett.*, vol. 108, no. 13, Mar. 2012, Art. no. 130503.

[12] Y.-L. Tang, H.-L. Yin, Q. Zhao, H. Liu, X.-X. Sun, M.-Q. Huang, W.-J. Zhang, S.-J. Chen, L. Zhang, L.-X. You, Z. Wang, Y. Liu, C.-Y. Lu, X. Jiang, X. Ma, Q. Zhang, T.-Y. Chen, and J.-W. Pan, "Measurement-device-independent quantum key distribution over untrustful metropolitan network," *Phys. Rev. X*, vol. 6, no. 1, Mar. 2016, Art. no. 011024.

[13] H.-L. Yin, T.-Y. Chen, Z.-W. Yu, H. Liu, L.-X. You, Y.-H. Zhou, S.-J. Chen, Y. Mao, M.-Q. Huang, W.-J. Zhang, H. Chen, M. J. Li, D. Nolan, F. Zhou, X. Jiang, Z. Wang, Q. Zhang, X.-B. Wang, and J.-W. Pan, "Measurement-device-independent quantum key distribution over a 404 km optical fiber," *Phys. Rev. Lett.*, vol. 117, no. 19, Nov. 2016, Art. no. 190501.

[14] H. Liu, W. Wang, K. Wei, X.-T. Fang, L. Li, N.-L. Liu, H. Liang, S.-J. Zhang, W. Zhang, H. Li, L. You, Z. Wang, H.-K. Lo, T.-Y. Chen, F. Xu, and J.-W. Pan, "Experimental demonstration of high-rate measurement-device-independent quantum key distribution over asymmetric channels," *Phys. Rev. Lett.*, vol. 122, no. 16, Apr. 2019, Art. no. 160501.

[15] Z. Zhao, Y. Zhao, Y. Li, F. Wang, X. Li, D. Han, and J. Zhang, "Service restoration in multi-modal optical transport networks with reinforcement learning," *Opt. Exp.*, vol. 29, no. 3, pp. 199–203, 2020.

[16] M. Lucamarini, Z. L. Yuan, J. F. Dynes, and A. J. Shields, "Overcoming the rate-distance limit of quantum key distribution without quantum repeaters," *Nature*, vol. 557, no. 7705, pp. 400–403, May 2018.

[17] L. Zhang, L. Tang, S. Zhang, Z. Wang, X. Shen, and Z. Zhang, "A self-adaptive reinforcement-exploration *Q*-learning algorithm," *Symmetry*, vol. 13, no. 6, p. 1057, Jun. 2021, doi: 10.3390/sym13061057.

[18] X.-T. Fang, P. Zeng, H. Liu, M. Zou, W. Wu, Y.-L. Tang, Y.-J. Sheng, Y. Xiang, W. Zhang, H. Li, Z. Wang, L. You, M.-J. Li, H. Chen, Y.-A. Chen, Q. Zhang, C.-Z. Peng, X. Ma, T.-Y. Chen, and J.-W. Pan, "Implementation of quantum key distribution surpassing the linear rate-transmittance bound," *Nature Photon.*, vol. 14, no. 7, pp. 422–425, Jul. 2020.

[19] Y. Cao, Y. Zhao, J. Li, R. Lin, J. Zhang, and J. Chen, "Hybrid trusted/untrusted relay-based quantum key distribution over optical backbone networks," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 9, pp. 2701–2718, Sep. 2021, doi: 10.1109/JSAC.2021.3064662.

[20] Y. Cao, Y. Zhao, J. Li, R. Lin, J. Zhang, and J. Chen, "Mixed relay placement for quantum key distribution chain deployment over optical networks," in *Proc. Eur. Conf. Opt. Commun. (ECOC)*, Dec. 2020, pp. 1–4.

[21] X. Zou, X. Yu, Y. Zhao, A. Nag, and J. Zhang, "Collaborative routing in partially-trusted relay based quantum key distribution optical networks," in *Proc. Opt. Fiber Commun. Conf. (OFC)*, 2020, pp. 1–3.

[22] X. Li, Y. Zhao, A. Nag, X. Yu, and J. Zhang, "Key-recycling strategies in quantum-key-distribution networks," *Appl. Sci.*, vol. 10, no. 11, p. 3734, May 2020, doi: 10.3390/app10113734.

[23] Y. Cao, Y. Zhao, C. Colman-Meixner, X. Yu, and J. Zhang, "Key on demand (KoD) for software-defined optical networks secured by quantum key distribution (QKD)," *Opt. Exp.*, vol. 25, no. 22, pp. 26453–26467, Oct. 2017.

[24] Y. Cao, Y. Zhao, Y. Wu, X. Yu, and J. Zhang, "Time-scheduled quantum key distribution (QKD) over WDM networks," *J. Lightw. Technol.*, vol. 36, no. 16, pp. 3382–3395, Aug. 15, 2018.

[25] A. Aguado, E. Hugues-Salas, P. A. Haigh, J. Marhuenda, A. B. Price, P. Sibson, J. E. Kennard, C. Erven, J. G. Rarity, M. G. Thompson, A. Lord, R. Nejabati, and D. Simeonidou, "Secure NFV orchestration over an SDN-controlled optical network with time-shared quantum key distribution resources," *J. Lightw. Technol.*, vol. 35, no. 8, pp. 1357–1362, Apr. 15, 2017.

[26] P. Sharma, V. Bhatia, and S. Prakash, "Efficient ordering policy for secret key assignment in quantum key distribution-secured optical networks," *Opt. Fiber Technol.*, vol. 68, Jan. 2022, Art. no. 102755, doi: 10.1016/j.yofte.2021.102755.

[27] E. Hugues-Salas, F. Ntavou, D. Gkounis, G. T. Kanellos, R. Nejabati, and D. Simeonidou, "Monitoring and physical-layer attack mitigation in SDN-controlled quantum key distribution networks," *J. Opt. Commun. Netw.*, vol. 11, no. 2, p. A209, Feb. 2019.

[28] F.-H. Xu et al. (Jul. 2021). *Network Coding in Trusted Relay Based Quantum Network*. [Online]. Available: http://individual.utoronto.ca/Tiger_Xu/Research_files/NCodingQKD.pdf.

[29] Y. Cao, Y. Zhao, X. Yu, and J. Zhang, "Multi-tenant provisioning over software defined networking enabled metropolitan area quantum key distribution networks," *J. Opt. Soc. Amer. B, Opt. Phys.*, vol. 36, no. 3, p. B31, Mar. 2019.

[30] Y. Cao, Y. Zhao, X. Yu, L. Cheng, Z. Li, G. Liu, and J. Zhang, "Experimental demonstration of end-to-end key on demand service provisioning over quantum key distribution networks with software defined networking," in *Proc. Opt. Fiber Commun. Conf. (OFC)*, 2019, pp. 1–3.

[31] J. Y. Cho, J.-J. Pedreno-Manresa, S. Patri, A. Seggev, J.-P. Elbers, H. Griesser, C. White, and A. Lord, "Demonstration of software-defined key management for quantum key distribution network," in *Proc. Opt. Fiber Commun. Conf. (OFC)*, 2021, pp. 1–3.

[32] Y. Ou, E. Hugues-Salas, F. Ntavou, R. Wang, Y. Bi, S. Yan, G. Kanellos, R. Nejabati, and D. Simeonidou, "Field-trial of machine learning-assisted quantum key distribution (QKD) networking with SDN," in *Proc. Eur. Conf. Opt. Commun. (ECOC)*, Sep. 2018, pp. 1–3, doi: 10.1109/ECOC.2018.8535497.

[33] Y. Cao, Y. Zhao, J. Li, R. Lin, J. Zhang, and J. Chen, "Multi-tenant provisioning for quantum key distribution networks with heuristics and reinforcement learning: A comparative study," *IEEE Trans. Netw. Service Manage.*, vol. 17, no. 2, pp. 946–957, Jun. 2020, doi: 10.1109/TNSM.2020.2964003.

[34] Y. Cao, Y. Zhao, J. Li, R. Lin, J. Zhang, and J. Chen, "Reinforcement learning based multi-tenant secret-key assignment for quantum key distribution networks," in *Proc. Opt. Fiber Commun. Conf. (OFC)*, 2019, pp. 1–3.

[35] X. Lin, G. Hou, W. Lin, and K. Chen, "Quantum key distribution in partially-trusted QKD ring networks," in *Proc. IEEE 3rd Int. Conf. Inf. Syst. Comput. Aided Educ. (ICISCAE)*, Sep. 2020, pp. 33–36, doi: 10.1109/ICISCAE51034.2020.9236910.

[36] Y. Zhao, K. Zhang, Q. Zhu, H. Wang, X. Yu, and J. Zhang, "Applications of machine learning in quantum key distribution networks," in *Proc. IEEE 6th Optoelectron. Global Conf. (OGC)*, Sep. 2021, pp. 227–229, doi: 10.1109/OGC52961.2021.9654412.

[37] C. H. Bennett, "Quantum cryptography: Public key distribution and coin tossing," in *Proc. IEEE Int. Conf. Comput. Syst. Signal Process.*, 1984, pp. 7–11.

[38] V. Scarani, A. Acín, G. Ribordy, and N. Gisin, "Quantum cryptography protocols robust against photon number splitting attacks for weak laser pulse implementations," *Phys. Rev. Lett.*, vol. 92, no. 5, Feb. 2004, Art. no. 057901.

[39] H. Singh, D. Gupta, and A. Singh, "Quantum key distribution protocols: A review," *IOSR J. Comput. Eng.*, vol. 16, no. 2, pp. 1–9, 2014.

[40] L. Salvail, M. Peev, E. Diamanti, R. Alléaume, N. Lütkenhaus, and T. Länger, "Security of trusted repeater quantum key distribution networks," *J. Comput. Secur.*, vol. 18, no. 1, pp. 61–87, Jan. 2010.

[41] T. Langer, "The practical application of quantum key distribution," Ph.D. thesis, Dept. Inf. Syst., Univ. Lausanne, Lausanne, Switzerland, 2013.

[42] C. Elliott, "Quantum cryptography," *IEEE Security Privacy*, vol. 2, no. 4, pp. 57–61, Jul./Aug. 2004, doi: 10.1109/MSP.2004.54.

[43] P. Schartner and S. Rass, "How to overcome the 'Trusted node model' in quantum cryptography," in *Proc. Int. Conf. Comput. Sci. Eng.*, 2009, pp. 259–262, doi: 10.1109/CSE.2009.171.

[44] H. Wang, Y. Zhao, X. Yu, Z. Ma, J. Wang, A. Nag, L. Yi, and J. Zhang, "Protection schemes for key service in optical networks secured by quantum key distribution (QKD)," *J. Opt. Commun. Netw.*, vol. 11, no. 3, pp. 67–78, Mar. 2019, doi: 10.1364/JOCN.11.000067.

[45] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "OpenFlow: Enabling innovation in campus networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Apr. 2008. [Online]. Available: http://doi.acm.org/10.1145/1355734.1355746.

[46] F. Karinou, H. H. Brunner, C.-H. F. Fung, L. C. Comandar, S. Bettelli, D. Hillerkuss, M. Kuschnerov, S. Mikroulis, D. Wang, C. Xie, M. Peev, and A. Poppe, "Toward the integration of CV quantum key distribution in deployed optical networks," *IEEE Photon. Technol. Lett.*, vol. 30, no. 7, pp. 650–653, Apr. 1, 2018, doi: 10.1109/LPT.2018.2810334.

[47] H. Wang, Y. Zhao, and A. Nag, "Quantum-key-distribution (QKD) networks enabled by software-defined networks (SDN)," *Appl. Sci.*, vol. 9, no. 10, p. 2081, May 2019, doi: 10.3390/app9102081.

[48] Y. Cao, Y. Zhao, J. Wang, X. Yu, Z. Ma, and J. Zhang, "SDQaaS: Software defined networking for quantum key distribution as a service," *Opt. Exp.*, vol. 27, no. 5, pp. 6892–6909, Mar. 2019, doi: 10.1364/OE.27.006892.

[49] F. Zeng, C. Wang, and S. S. Ge, "A survey on visual navigation for artificial agents with deep reinforcement learning," *IEEE Access*, vol. 8, pp. 135426–135442, 2020, doi: 10.1109/ACCESS.2020.3011438.

[50] K. R. Raghunandan, A. Ganesh, S. Surendra, and K. Bhavya, "Key generation using generalized Pell's equation in public key cryptography based on the prime fake modulus principle to image encryption and its security analysis," *Cybern. Inf. Technol.*, vol. 20, no. 3, pp. 86–101, Sep. 2020, doi: 10.2478/cait-2020-0030.

[51] R. Parkinson, "Traffic engineering techniques in telecommunications," Infotel Syst. Corp., Richmond, VA, USA, Tech. Rep., 2002.

[52] Q. Zhang, Y. Liu, X. Yu, Y. Zhao, and J. Zhang, "Topology-abstraction-based protection scheme in quantum key distribution networks with partially trusted relays," *Photonics*, vol. 9, no. 4, p. 239, Apr. 2022, doi: 10.3390/photonics9040239.

[53] M. Campagna et al., "Quantum safe cryptography and security," Sophia Antipolis, France, White Paper, 2012. [Online]. Available: https://www.etsi.org/images/files/ETSIWhitePapers/QuantumSafeWhitepaper.pdf

[54] M. Poev et al., "The SECOQC quantum key distribution network in Vienna," *New J. Phys.*, vol. 11, no. 7, Jul. 2009, Art. no. 075001, doi: 10.1088/1367-2630/11/7/075001.

[55] C. Yu, J. Lan, Z. Guo, and Y. Hu, "DROM: Optimizing the routing in software-defined networks with deep reinforcement learning," *IEEE Access*, vol. 6, pp. 64533–64539, 2018, doi: 10.1109/ACCESS.2018.2877686.

[56] D. Elkouss, J. Martinez-Mateo, A. Ciurana, and V. Martin, "Secure optical networks based on quantum key distribution and weakly trusted repeaters," *J. Opt. Commun. Netw.*, vol. 5, no. 4, pp. 316–328, Apr. 2013, doi: 10.1364/JOCN.5.000316.

[57] Y. Cao et al., "Long-distance free-space measurement-device-independent quantum key distribution," *Phys. Rev. Lett.*, vol. 125, no. 26, Dec. 2020, Art. no. 260503.

[58] J.-P. Chen, C. Zhang, Y. Liu, C. Jiang, W. Zhang, X.-L. Hu, J.-Y. Guan, Z.-W. Yu, H. Xu, J. Lin, M.-J. Li, H. Chen, H. Li, L. You, Z. Wang, X.-B. Wang, Q. Zhang, and J.-W. Pan, "Sending-or-not-sending with independent lasers: Secure twin-field quantum key distribution over 509 km," *Phys. Rev. Lett.*, vol. 124, no. 7, Feb. 2020, Art. no. 070501.

[59] A. I. Nurhadi and N. R. Syambas, "Quantum key distribution (QKD) protocols: A survey," in *Proc. 4th Int. Conf. Wireless Telematics (ICWT)*, Jul. 2018, pp. 1–5.

[60] Y. Cao, Y. Zhao, J. Wang, X. Yu, Z. Ma, and J. Zhang, "Cost-efficient quantum key distribution (QKD) over WDM networks," *J. Opt. Commun. Netw.*, vol. 11, no. 6, pp. 285–298, Jun. 2019.

OMAR SHIRKO (Graduate Student Member, IEEE) received the M.S. degree in information technology (computer science) from the National University of Malaysia, Selangor, Malaysia. He is currently pursuing the Ph.D. degree in information system engineering with Erbil Polytechnic University, Erbil, Kurdistan Region, Iraq. Since 2012, he has been working as a University Lecturer with Erbil Polytechnic University. His research interests include quantum cryptography, artificial intelligence, machine learning, reinforcement learning, and deep learning.

SHAVAN ASKAR (Senior Member, IEEE) received the B.Sc. (Hons.) and M.Sc. degrees from the Control and Systems Engineering Department, University of Technology, Baghdad, in 2001 and 2003, respectively, and the Ph.D. degree in electronic systems engineering from the University of Essex, U.K., in 2012. He is currently an Associate Professor of computer networks. He is also the CEO of Arcella Telecom. He has more than 60 scientific research papers, some of his papers were published in highly prestigious conferences, such as OFC and ECOC, and high-impact-factor (3.58) journals, such as *Optics Express*. His research interests include the Internet of Things, software-defined networks, optical networks, and 5G.