INTERNATIONAL TELECOMMUNICATION UNION

TELECOMMUNICATION STANDARDIZATION SECTOR

STUDY PERIOD 2017-2020

SG13-TD556/WP3

STUDY GROUP 13

Original: English

Question(s): 16/13 Virtual, 1-12 March 2021

TD

Source: Editors

Title: Draft Supplement ITU-T Y.supp.QKDN-mla : "Quantum Key Distribution Networks - Applications of Machine Learning"

Purpose: Proposal

Contact:	Qingcheng Zhu Beijing University of Posts and Telecommunications. China	Tel: +86-18611519588 E-mail: qingcheng@bupt.edu.cn
Contact:	Yongli Zhao Beijing University of Posts and Telecommunications. China	Tel: +86-10-61198108 E-mail: yonglizhao@bupt.edu.cn
Contact:	Xiaosong Yu Beijing University of Posts and Telecommunications. China	Tel: +86-10-61198108 E-mail: xiaosongyu@bupt.edu.cn
Contact:	Zhangchao Ma CAS Quantum Network Co., Ltd. China	Tel: +86-10-83057625 E-mail: mazhangchao@casquantumnet.com
Contact:	Junsen Lai Ministry of Industry and Information Technology (MIIT). China	Tel: +86-10-62300592 E-mail: laijunsen@caict.ac.cn

Keywords: Quantum key distribution (QKD); QKD networks; machine learning (ML); applications.

Abstract: This contribution includes the detailed content for the new established Technical Report ITU-T TR.QKDN-mla "Applications of Machine Learning in Quantum Key Distribution Networks" after December 2020 SG13RGM Q16/13 meeting.

Summary

The initial draft Technical Report TR.QKDN-mla "Applications of Machine Learning in Quantum Key Distribution Networks" was created in the December 2020 SG13RGM Q16/13 meeting and the base line document was included in TD517/WP3. The draft was agreed to be converted as new Supplement to Y.3800-series - Quantum Key Distribution Networks - Applications of Machine Learning in the March 2021 SG13 Q16/13 meeting. This output document includes the detailed content for the Supplement. The updated texts are included.

Annex I

Draft Supplement to ITU-T Y-series Recommendations

ITU-T Y.3800-series - Quantum Key Distribution Networks - Applications of Machine Learning

1. Scope

This Supplement studies the applications of machine learning (ML) in quantum key distribution network (QKDN).

In particular, the scope of this draft Supplement will include:

- Overview of ML applications in QKDN;
- Applications of ML in QKDN at the quantum layer;
- Applications of ML in QKDN at the key management layer;
- Applications of ML in QKDN at the control and management layers.

2. References

[ITU-T Y.3800] Recommendation ITU-T Y.3800 (2019), Framework for Networks to support Quantum Key Distribution.

[ITU-T Y.3801] Recommendation ITU-T Y.3801 (2020), Functional requirements for quantum key distribution networks.

[ITU-T Y.3802] Recommendation ITU-T Y.3802 (2020), Functional architecture of the Quantum Key Distribution network.

[ITU-T Y.3803] Recommendation ITU-T Y.3803 (2020), Key management for quantum key distribution network.

[ITU-T Y.3804] Recommendation ITU-T Y.3804 (2020), Control and Management for Quantum Key Distribution Network.

[ITU-T Y Suppl. 55] Supplement to ITU-T Y.3170-series (2019), Machine learning in future networks including IMT-2020: use cases.

[ITU-T Y.3170] Recommendation ITU-T Y.3170 (2018), Requirements of machine learning based QoS assurance for IMT-2020 network.

[ITU-T Y.3172] Recommendation ITU-T Y.3172 (2019), Architectural framework for machine learning in future networks.

3. Terms and definitions

3.1. Terms defined elsewhere

[Editor's note: Please check the necessity of adding several terms and definitions.]

This Supplement uses the following terms defined elsewhere:

3.1.1. Quantum key distribution (QKD) [b-ETSI GR QKD 007]: Procedure or method for generating and distributing symmetrical cryptographic keys with information theoretical security based on quantum information theory.

3.1.2. Quantum key distribution network (QKDN) [ITU-T Y.3800]: A network comprised of two or more quantum key distribution (QKD) nodes connected through QKD links.

NOTE – A QKDN allows sharing keys between the QKD nodes by key relay when they are not directly connected by a QKD link.

3.1.3. Machine learning (ML) [ITU-T Y.3172]: Processes that enable computational systems to understand data and gain knowledge from it without necessarily being explicitly programmed.

NOTE 1 – Definition adapted from [b-ETSI GR ENI 004].

NOTE 2 – Supervised machine learning and unsupervised machine learning are two examples of machine learning types.

3.2. Terms defined in this Recommendation

This chapter defines all the terms used in this Supplement.

TBD

4. Abbreviations and acronyms

This chapters describes all the abbreviations and acronyms used in the Supplement.

TBD

5. Conventions

None.

6. Overview

Quantum key distribution (QKD) technology can achieve information theoretical secure with one-time pad encryption. Machine learning (ML) provides systems the ability to automatically learn and improve from experience without being explicitly programmed. There are some applications of ML in telecommunication networks, such as traffic prediction and fault prediction. There is increasing interest in the field in applying ML to improve the performance of quantum communication networks.

The ML module has the functions including the data processing, data storing, ML model selecting, ML model training. The ML module is in the QKDN controller at the QKDN control layer. The ML module collects input data from other layers through ML-related reference points, which is for the further data processing. The ML module gets access to the service data and quantum layer information from the key management layer. It can also achieve the QKDN management information at the QKDN management layer. The output of the ML module is the output of the trained ML model, which can be applied to other layers. Based on the academic and industrial advances, the supplement studies the applications of ML in QKDN. The supplement clarifies the applications of ML at the quantum layer, at the key management layer and at the management and control layers including the use case description, the use case analysis and, the benefits and impact.

[Editor's note: Please describe the ML module clearly with a diagram.]

7. Applications of ML in QKDN at the quantum layer

7.1. Introduction

The applications of ML at the quantum layer represent applying the ML at the quantum layer and improving the performance of the quantum layer such as the quantum channel performance. For example, by selecting the quantum channel with the best transmission performance based on the prediction results can reduce the influence of noise and improve the transmission quality of the quantum signal.

7.2. Use case QL01: ML-based quantum channel performance prediction

Use case title

ML-based quantum channel performance prediction

[Editor's note: the related description and analysis should be revised according to the use case title.]

Use case description

(1) Background

Channel performance and transmission environment in the Quantum Layer is crucial for bringing the implementation and commercialisation of quantum networks into reality. During the transmission of high-intensity quantum-encoded photons, the main challenge is the photon-induced noise falling in the quantum channel (Ch-QKD) that deteriorates the Ch-QKD quality. Supervised machine learning (ML) method is recommended to estimate the Ch-QKD performance (noise, Optical Signal Noise Ratio-OSNR) in the presence of Ch-QKD in various quantities, spectrum allocations, launch power and channels spacing.

(2) Issue

QKD is the most widely researched branch in quantum communication proposals to achieve secure key transmission. Benefited from the laws of quantum mechanics, the possible eavesdropping attacks will be detected during the QKD process. However, low secure key rate (SKR) is a significant challenge for the practical QKD. SKR is often related to parameters such as single photon detector (SPD) photon detection output count and quantum channel bit error rate, and will change with the attenuation of the channel. Inevitable channel noise during photon transmission will reduce the security of the communication process. Being directly related to quantum bit-error ratio (QBER), optical signal-to-noise ratio (OSNR) has become one of the most crucial monitoring parameters. Recently, ML-based techniques have been applied to optical communication to predict OSNR. To avoid the drop in SKR caused by channel noise, it is essential to adopt ML-based technique to perform OSNR prediction.

(3) Role of ML in QKDN

During the QKD process, the noise in the quantum channel will cause the transmission rate of the quantum signal to decrease. As the quantum OSNR decreases, the QBER begins to rise. When the QBER approaches the security threshold, the SKR began to decline sharply. The ML-based quantum channel OSNR prediction solution is to predict the OSNR of the quantum state optical signal in the quantum channel by combining the SPD, photon detection output counter, the quantum channel error rate and according to the coding rate under different channel noise environments. Based on the predicted OSNR, actions can be taken in advance to improve the channel environment and reduce unnecessary loss caused by SKR decreases. Fig 7.1 shows the diagram of ML-based quantum channel OSNR prediction.

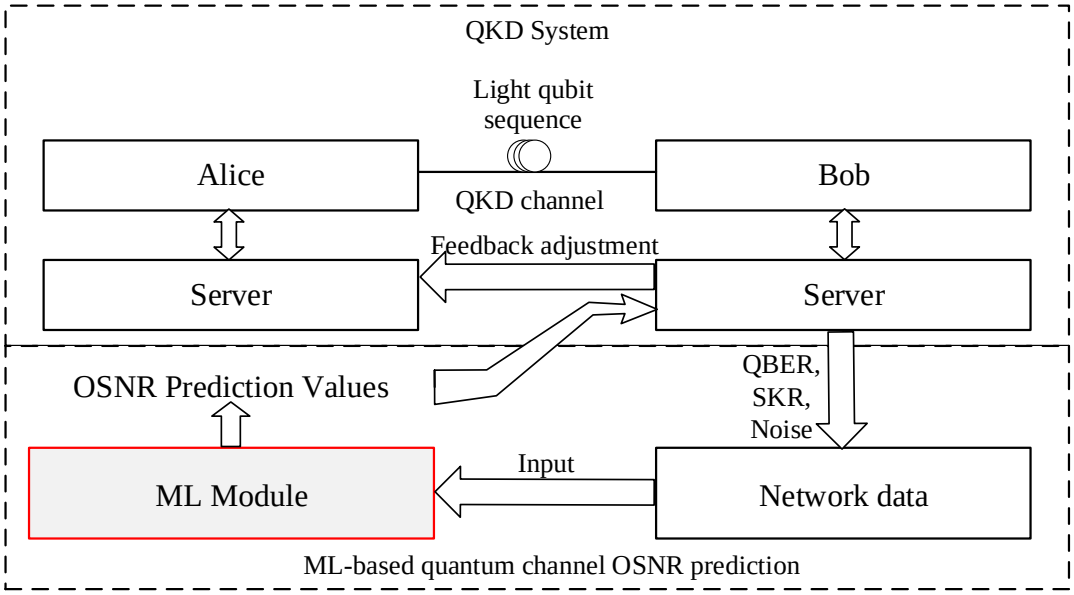


Fig 7.1. Diagram of ML-based quantum channel performance prediction

Use case analysis

- Analysis related to data collection

  1) The ML module collects quantum signals data under the influence of different noises through the quantum channel in the quantum layer.
  2) The collected quantum signals data includes the intensity information of the signal (e.g. bit error rate of quantum channel, the SPD photon detection output counter, code formation rate under different noise environment).

- Analysis related to data storage and processing

  1) It supports storage of data used for analytics. A channel database in the QKDN quantum layer stores collected data and possibly stores predictions.
  2) It supports the accuracy of OSNR prediction under different channel noise environments.

- Analysis related to application of ML output

  1) The ML output is applied in the parameters that reflect the QKD channel performance and OSNR to be predicted
  2) The QL is able to implement intelligent signal analysis function and applied in ensuring robust network operation according to ML output.

Benefits and impact

ML-based quantum channel OSNR prediction method will predict the signal-to-noise ratio of the quantum state optical signal in the quantum channel according to the encoding rate under different channel noise environments. The predicted signal-to-noise ratio make the quantum channel in the optimal performance state in real time, and measures can be taken in advance to improve the channel environment and reduce unnecessary losses caused by reducing the signal-to-noise ratio.

7.3. Use case QL02: ML-based QKD system parameter optimization

Use case title

ML-based QKD system parameter optimization

Use case description

(1) Background

For a practical quantum key distribution (QKD) system, due to the low efficiency of the basis-sift factor, this will lead to the low efficiency of key generation. Therefore, in order to obtain better performance in terms of key generation rate and secure transmission distance, it is necessary to optimize the full parameters of the QKD system. There are two sets of parameters: the intensities of signal and decoy state and the probabilities to choose different intensities and bases, as well as the probability that Bob measures the incoming pulse with Z basis. In other words, in order to make the QKD system at its best performance, we have to optimize the full parameters of the QKD system over multiple dimensions.

(2) Issue

In a practical QKD system, considering the influence of limited communication time, the selection of intensities and the probability of sending these intensities are essential to obtain the optimal quantum key rate. The QKD network connects multiple devices, there are a large number of QKD requests, and the QKD system parameter optimization is performed in multiple dimensions, all of which require considerable computing power. Insufficient computing power of the QKD network controller will cause the QKD system to either wait for an optimization off-line (and suffer from delay) or use suboptimal or even unoptimized parameters in real time, which will reduce the efficiency of the basis-sift factor. This will bring huge computational challenges to the controller of a quantum network with multiple pairs of users (where real-time optimization might simply be infeasible for even a moderate number of connections). At the same time, when the gain and quantum bit error rate change with the environment, the parameters of the QKD system also need to be re-optimized, so that the QKD system always maintains the best performance. Therefore, how to quickly and accurately optimize the parameters of the QKD system is a difficult task.

Traditionally, parameter optimization has relied on brute-force search, local search or powerful global search algorithms. These algorithms are computationally intensive and slow on low-power platforms (this will increase system latency), requiring a lot of time and hardware resources. It means that real-time QKD system optimization is not realistic. The ML technology can find rules from a large amount of training data and quickly give the optimal QKD system parameters.

Role of ML in QKDN

The ML-based QKD system parameter optimization solution is to pre-execute QKD system parameter optimization before key generation. Since each layer of the neural network (NN) is composed of many neurons, after accepting the input from the previous layer and calculating the activation, it outputs the signal to the next layer. And the NN is a highly flexible and robust structure that can be used in a wide range of scenarios where such mappings between two finite input or output vectors exist. Therefore, input the optimal parameters obtained by NN technology into the system.

Fig. 7.2 shows the diagram of ML-based QKD system parameter optimization. Firstly, randomly sample the input data space to pick a random combination of physical parameters which cannot be controlled by the users and use a local search algorithm to calculate their corresponding optimization parameters which can be adjusted by the users. The obtained physical parameters are input into NN trainer. After the multi-layer NN, N sets of prediction parameters are output. By comparing the key rate obtained by the classical algorithm with the key rate based on the prediction parameters, the comparison result is feedback to the trainer to complete the training. Secondly, when the QKD system needs parameter optimization, input the real-time data, output the optimization parameters after passing through the NN model. Finally, inputting the result into the QKD system to complete parameter optimization.

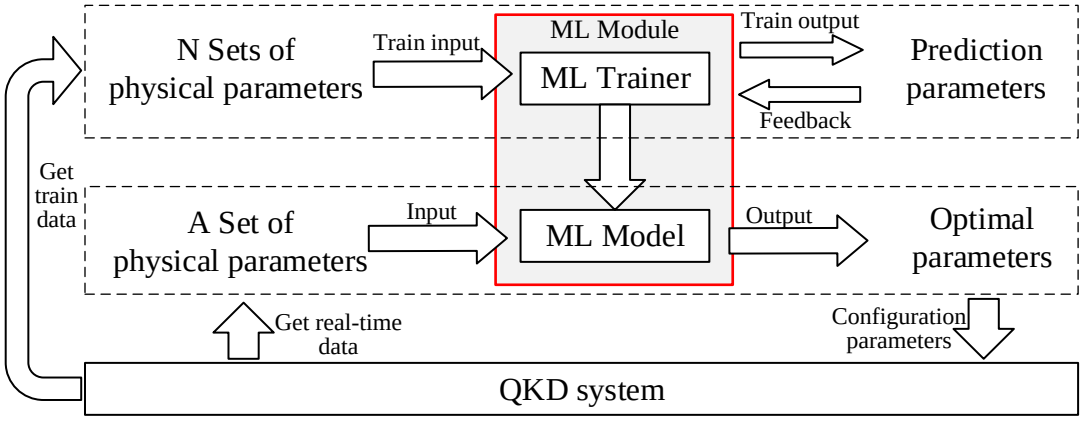


Fig. 7.2 Diagram of ML-based QKD system parameter optimization

Use case analysis

- Analysis related to data collection

  1) It randomly samples the input data space to pick a random combination of physical parameters and use a local search algorithm to calculate their corresponding optimization parameters.
  2) It collects the optimal parameters (e.g., the choice of signal intensities and the probabilities of sending them) by the classical algorithm in quantum layer.
  3) It collects the physical parameters (e.g., distance between Alice and Bob, the detector efficiency, the dark count probability, the basis misalignment, the error-correction efficiency, the number of signals) in quantum layer.

- Analysis related to data storage and processing

  1) It supports simple scaling and normalization of input data
  2) It supports real-time prediction which aims to predict optimal parameter.
  3) It supports retraining the ML model after the update of datasets.

- Analysis related to application of ML output

  1) The ML output is applied to compare with previous parameter settings.
  2) The ML output is applied before quantum key generation.

Benefits and impact

The ML-based QKD system parameter optimization solution will optimize the QKD system quickly and accurately in a more effective and accurate way on the low-power platform, based on the real-time changing environment, so as to make the QKD system in the optimal performance state in real time.

7.4. Use case QL03: ML-based remaining use life (RUL) prediction of components in QKD system

Use case title

ML-based remaining use life (RUL) prediction of components in QKD system

Use case description

(1) Background

In the quantum key distribution (QKD) system, the life cycle of the QKD equipment at the quantum layer is essential to the normal operation of the QKD system. With the extension of the working hours of QKD equipment, the phenomenon of aging equipment will appear, which will cause the equipment

life to end suddenly and have a great influence on the service request in the QKD system. Machine learning (ML) has powerful data processing capabilities. It can build a training model to predict the RUL of the QKD equipment by collecting the working hours of equipment or system, the operating conditions information and component life cycle data. Moreover, it also obtains other parameters which can be used to train to improve the generalization ability of the model, and it can ensure the true life and accuracy of the RUL of the QKD equipment.

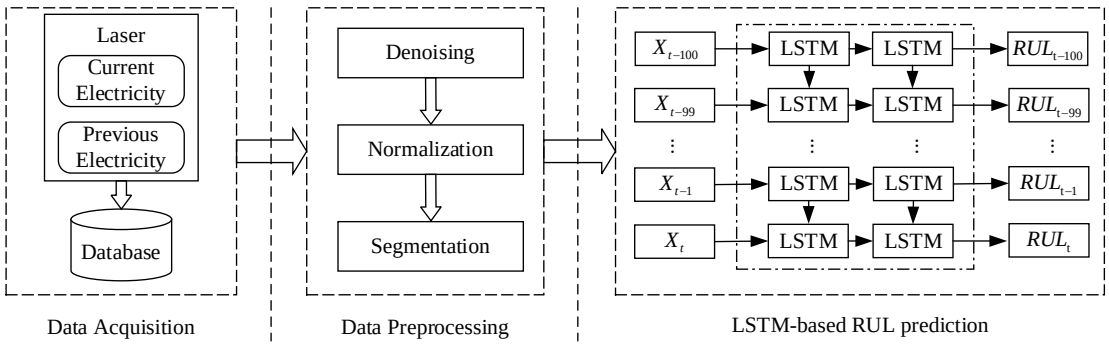
(2) Issue

Considering the double influence of external environment and internal factors, the life cycle of QKD equipment is an unavoidable problem. As we all know, with the continuous use of components, the working performance status and accuracy of the components will decrease over time, which will affect the reliability of the generated key resources and the security of service request encryption in the QKD network. In the QKD system, the QKD equipment includes pulsed light source, decoy state modulation module, quantum state modulation module, adjustable optical attenuator, random number generator, single photon detector (SPD) etc.

Considering the time uncertainty of component damage and the expensive price of QKD equipment, the cost of finding damaged components and replacing them in time is huge. Therefore, it is very important and necessary to make an accurate prediction of the RUL of QKD equipment. ML can collect related data under various equipment status, and then perform feature engineering on the data. Finally, ML selects appropriate machine learning algorithms for model training and completes the RUL prediction of QKD equipment.

(3) Role of ML in QKDN

Lasers are widely used in various fields, including optical communications, military and medical applications. However, the physical phenomenon of laser degradation will appear over time and we don't know when it will be damaged. Therefore, as one of many components in QKD system, understanding the true life of the laser is great meaningful for the QKD system. Considering the laser degradation modes are observed at different time scales: gradual degradation could extend to several hundreds of hours, and catastrophic degradation appears after many hours of normal operation, an accurate prediction requires the combination of the recent measurements representing the latest change and the historical sensor measurements jointly modelling the degradation tendency. In order to ensure the reliability and availability of the laser, the RUL prediction of the laser can be carried out through Long Short-Term Memory (LSTM) methods, because LSTM networks are well-suited to classifying, processing and making predictions based on time series data. So, the laser prediction process based on LSTM mainly includes three stages: data acquisition, data pre-processing and RUL prediction, as shown in Fig. 7.3.



X_t : Each run-to-failure data sequence is a univariate time series.
RUL_t : For each data sample, the LSTM model predicts the RUL at that sample time.

Fig. 7.3 Diagram of ML-based laser RUL prediction

Use case analysis

- Analysis related to data collection

1) It collects different parameters (e.g., temperature, power, electricity etc.) of components in QKD system.
2) It collects relevant historical data for model training.
3) It collects the related values periodically by using sensor.

- Analysis related to data storage and processing

1) It supports storage of the measured values used for analytics.
2) It supports the data processing such as normalization and segmentation.

- Analysis related to application of ML output

1) The ML output is applied to the assessment of component status.
2) The ML output is applied to the equipment performance analysis.

Benefits and impact

The ML-based RUL prediction of components in QKD system can more accurately estimate the remaining use life of the components, which greatly improves the operability of the components before the failure of the components in QKD system, and provides a guarantee for the normal operation of QKD system.

8. Applications of ML in QKDN at the key management layer

8.1. Introduction

The applications of ML at the key management layer represent applying the ML at the key management layer and improving the key management efficiency. For example, by applying the ML in the management of quantum key distribution pool, the key utilization of QKDN will be improved.

8.2. Use case KM01: ML-based key formatting

Use case title

ML-based key formatting

Use case description

(1) Background

The key manager (KM) is recommended to format keys where necessary for internal purposes or for key supply or key relay, including combining or splitting where lengths are not appropriate, which is stated in the recommendation ITU-T Y.3803. There are different key formats for different security requirements of services and different encryption algorithms (e.g. OTP, AES-512, AES-256, AES-128). To maintain interconnectivity and expandability in the QKDN, proper key format for key data with added metadata containing various types of information needs to be solved.

(2) Issue

The key formatting is necessary before storing keys. The lengths of the acquired QKD-key files may differ from each other. Therefore, as recommended by the Req_KM 3 in [ITU-T Y.3801], the KM agent re-formats (combines or splits) the QKD-keys into keys of a prescribed unit length, and then temporarily store them in a buffer. However, the process of formatting and storing keys has neglected the key requirements of services. If no keys in the key store have the same format that a service require, the keys need to be re-formatted before the key supply. The key synchronization is needed after formatting or re-formatting keys, which may introduce the time cost and risk of key synchronization failure. Hence, through pre-executing key formatting with aware of service requirements, there is much optimization space in terms of time cost and risk of key synchronization failure during the key consumption. The stored keys may have different unit lengths and formats. The

number of keys with definite formats is affected by the service characteristics. The service characteristics include the service arriving time, the service duration time, the service security levels and the size of service data which needs encryption with keys. Note that, the service security level is used to describe the security requirement of a service.

For that the services dynamically arrive and have various types, the amount of service data is large. The traditional way such as using expert systems to predict the statistical service characteristics is inefficient. The ML technique has the ability to find the rules from a large amount of service data and predict the outcomes about service characteristics.

(3) Role of ML in QKDN

The ML-based key formatting solution is to guide the key formatting before storing keys with the awareness of the service characteristics. Fig. 8.1 shows the diagram of ML-based key formatting. A large amount of service information during a certain period is input into the ML module for training. The trained ML module is able to predict statistical service characteristics. According to the service characteristics output by the ML module, the key manager (KM) operates key formatting and store keys in the key store for the future key supply. As for the ML models, the prediction models such as deep learning algorithms and Elman neural network can be applied.

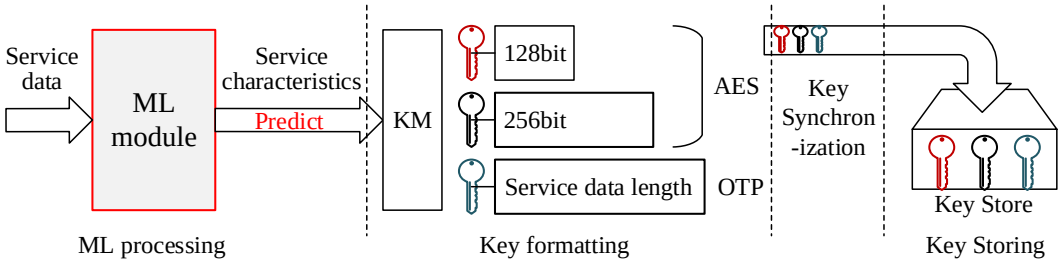


Fig. 8.1 Diagram of ML-based key formatting

Use case analysis

- Analysis related to data collection

  1) The KM collects the service data from the service layer. The service data is collected continuously for updating the prediction models in order to improve in real time the accuracy and effectiveness of the prediction models.
  2) The collected service data includes the service characteristics (e.g. the service arriving time, the service duration time, the service security levels and the size of service data which needs encryption with keys).

- Analysis related to data storage and processing

  1) It supports storage of data used for analytics. A database in the QKDN control layer stores collected data and possibly stores predictions.
  2) It supports real-time prediction which aims to predict service characteristics at cryptographic application.
  3) It supports predictions at different time granularity, such as real-time predictions (user activity), short-term predictions (user group activity) and long-term predictions (large-scale activity).

- Analysis related to application of ML output

  1) The ML output is applied in the process of key formatting before storing keys.
  2) The KM is able to operate key formatting according to the ML output.

Benefits and impact

The ML-based key formatting solution will reduce the time cost and the risk of key synchronization failure during the key consumption by guiding the key formatting with the awareness of service characteristics.

8.3. Use case KM02: ML-based key storage management

Use case title

ML-based key storage management

Use case description

(1) Background

In the research of QKD optical network, the key pool (KP) is usually used as a storage device, in which a continuous stream of generated keys is stored. When the security service requirements come, the corresponding number of keys can be directly obtained from the KP. With the increasing of user services, it is more and more necessary to control the key storage, so as to realize the reasonable scheduling and efficient utilization of channel resources and key resources. Therefore, it is suggested to evaluate the health state of the resource storage in the key pool to manage the key manager (KM) layer's key storage, so as to guide the optimal selection of the strategy. In the key management layer, KM is responsible for receiving and managing the quantum key generated by the QKD module of the quantum layer, relaying the key under the control of the QKDN control layer, and providing the key to the application layer. KM consists of key management agent (KMA), key supply agent (KSA) and KM control and management unit.

(2) Issue

QKD service cannot be carried out well with many factors, including insufficient or redundant key resources in the KP, long storage time, strong jitter and so on. In addition, the key demand of user service changes dynamically with the actual situation. However, the existing traditional solutions are difficult to accurately perceive the actual needs of the business. When the key requirement is greatly reduced or increased, the key supply cannot be dynamically adjusted, which leads to unnecessary waste and even security problems. Therefore, there is room for optimization in the forecast of business demand.

(3) Role of ML in QKDN

The ML-based key storage management solution is to reasonably predict the business demand before key supply. As shown in Fig. 8.2, the ML model has known a large amount of training data, through the classification algorithm to establish the model, compare the results and constantly adjust the model, and then accurately predicts the key requirements of the service. KM layer manages keys in the key pool according to the predicted value to ensure reasonable scheduling, efficient utilization of resources and avoid problems such as long key storage time or excessive jitter. The KMA module manages the key life cycle, which is to archive keys or to destroy the keys which have been stored for a long time. The KM control and management unit feeds back key generation status information to the QKDN controller, so as to inform the start or stop of the key generation process. Finally, the KSA module provides keys to the cryptographic application on demand.

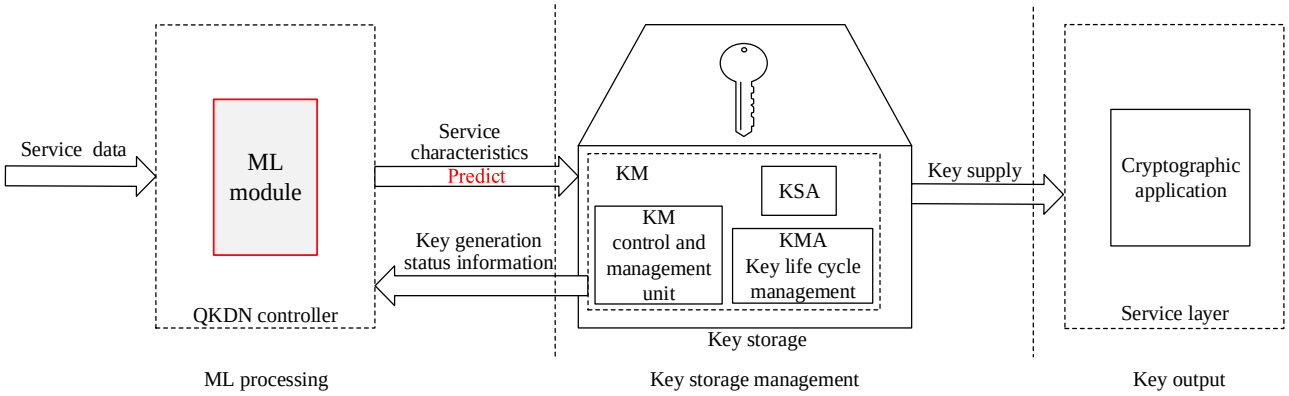


Fig. 8.2 Diagram of ML-based key storage management

Use case analysis

- Analysis related to data collection
  1) The KM collects data from the service layer. Constantly collect business demand data for updating the forecast model to improve the accuracy and effectiveness of the forecast model in real time.
  2) It collects business requirement data in KSA. (e.g. service type, security level, required key quantity).
- Analysis related to data storage and processing
  1) It supports the storage of a large amount of training data, and the analysis data is stored in the QKDN control layer.
  2) It supports real-time flexible classification methods, and the ML model can be retrained after data updates.
- Analysis related to application of ML output
  1) The ML output is applied in KMA to control the key life cycle.
  2) The ML output is applied in the KM control and management unit to feedback the state of key generation to the QKDN controller.

Benefits and impact

ML-based key storage management solution will guide the selection of the optimal path for key distribution, and realize the reasonable scheduling and efficient utilization of key resources.

8.4. Use case KM03: ML-based anomaly detection

Use case title

ML-based anomaly detection

[Editor's note: Please revise the description according to the use case title and clarify the anomaly detection at the key management layer.]

Use case description

(1) Background

A cryptographic application in the service layer sends a KSA a key request. The key request from cryptographic application may include information on a required security level, depending on key supply service policy, etc. As required by the Req_KM 10 in [ITU-T Y.3801], the KSA receives key requests from authorized cryptographic applications via a key supply interface at the reference point Ak. However, traditional cryptographic applications authentication processing can't effectively detect

mass attacks, such as DoS attack which uses reasonable service requests to gain excessive QKD network resources and results in an inability to access them by legitimate users. Therefore, the advantage of ML-based authorizing cryptographic applications is helping key manager distinguish the source nodes and address identity-based attacks such as spoofing and Sybil attacks well.

(2) Issue

The control of reception of key request(s) by the KSA can be supported by the QKDN controller, especially when key requests are sent from multiple cryptographic applications. The KSA then authenticates the cryptographic application by an appropriate means. Their certificate can be issued by an access control function of the QKDN controller, which manages an access control repository of registered functional components including cryptographic applications and KSAs. However, the traditional authentication processing for cryptographic applications can't effectively detect mass attacks. Hence, ML-based abnormal monitoring applied into authorizing cryptographic applications can help key manager distinguish the source nodes and address identity-based attacks such as spoofing and Sybil attacks well.

Role of ML in QKDN

The diagram of ML-based authorizing cryptographic applications is shown by Fig. 8.3. First of all, a cryptographic application in the service layer sends a key request, and then the ML module observes the current key request data from the cryptographic application and takes an action at each time step to monitor abnormal key request data. What's more, the ML module outputs judged result of normal request or abnormal request to KSA which authenticates the cryptographic application. Finally, the KSA supplies keys for the authenticated normal cryptographic application. In regard to ML models, ML algorithms supporting classification such as artificial neural network (ANN), recurrent neural network (RNN) can be applied.

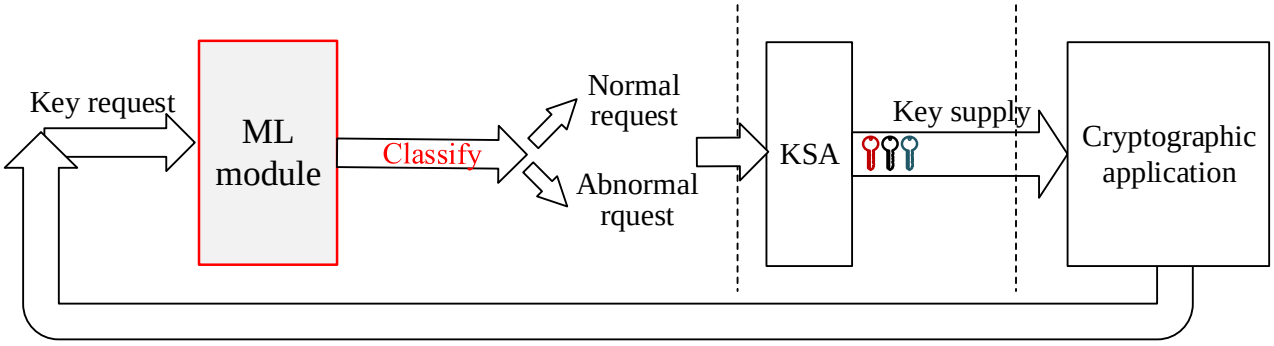


Fig. 8.3 Diagram of ML-based authorizing cryptographic applications

<u>Use case analysis</u>

- Analysis related to data collection

  1) It collects the key requirements of services arriving in a definite long period from the cryptographic application.
  2) It collects the requested information (e.g., key length, key amount, node pair names or IDs, KSA-key ID, and the security level of key) from cryptographic application to KSA.

- Analysis related to data storage and processing

  1) It supports storage of key data and metadata in KSA.
  2) It supports storage of data used for analytics.
  3) It supports retraining the ML model after the update of datasets.

- Analysis related to application of ML output

1) The ML output is applied in KSA.
2) The ML output is applied before the key supply, key relay or other processes to consume keys.

Benefits and impact

ML-based anomaly detection is to improve abnormal detection performance, and consequently to achieve considerably enhanced authentication accuracy.

9. Applications of ML in QKDN at the control and management layers

9.1. Introduction

The applications of ML at the control and management layers represent applying the ML at the control and management layers and improving the QKDN management and control efficiency. For example, by applying the ML in the routing selection, the QKDN resource utilization will be improved; by applying the ML in the fault diagnosis, the characteristics of the QKDN will be established more accurately and fault diagnosis in the QKDN will be performed more effectively.

9.2. Use case CML01: ML-based data collection and data pre-processing

Use case title

ML-based data collection and data pre-processing

Use case description

(1) Background

Data collection refers to the process of gathering a system's operation information (principal) prepared for the analyzing such as fault diagnose. The consumer is metadata. According to the data collection in QKDN, the management layer collects the information through the reference point of each layer and detect anomaly, then it makes decisions on the faults of layers for healing actions. Nonetheless, ordinary methods of data collection are not always adaptive, specifically in QKDNs with multi-source data which is heterogeneous. Therefore, the data collection and pre-processing is to categorize and aggregate data collected from reference points to help training the learning machine.

(2) Issue

With multi-source data transferred from reference point of each layer to the management layer, it needs to be processed in an appropriate manner. In the management layer, data processing is consisted of data collection, data pre-processing, and data analyzing. It contains configuration status, network topology, inventory resources and fault records of each layer. With respect to the functions of management layer, specific functions are sorted by three parts including quantum layer management (QLM) functions, key management layer management (KMLM) functions, and QKDN control layer management (QCLM) functions, which is stated in the recommendation ITU-T Y.3804 “Quantum key distribution networks – Control and management”.

- The QLM data collection focus on the QKD performance, QSNR, key generation rates, and parameters of QKD equipment provided by the quantum layer.
- The KMLM data collection is consisted of available amounts of keys in KM for key relay, key supply and key life cycle management.
- The QCLM data collection is mainly about routing and rerouting information recorded in the event log, network topology, and status of resources allocation.

With these data, the management layer makes an analysis for the fault diagnose. However, the data collected from QKDN is multi-sourced and heterogeneous which is not always suitable for the classical approaches of pre-processing. Such as the unexpected noise and redundancy raised in the collected metadata. Hence, ML can construct a data processing model that categorizes and aggregates data into understandable, unified and easy-to-use structures which is more practical in data analysis.

(3) Role of ML in QKDN

The ML-based data collection and pre-processing solution is to collect multi-source, heterogeneous QKDN metadata and transformed it into understandable, unified and easy-to-use structures of data for analysis. Fig. 9.1 shows the diagram of ML-based data collection and pre-processing. The metadata from quantum layer, key management layer and QKDN control layer are input into the management layer for pre-processing. According to the output by the ML module in metadata processing and data enhancement, the management layer makes an analysis and predicts the faults.

During the period of metadata processing, the collected data contains unexpected noise and redundancy. ML module is used to identify the derivative correlations between event logs and abstract the fault information. Then it generates a data set to train the ML module for further metadata processing.

During the period of data enhancement, shortage of specific data makes the metadata collection insufficient to be analyzed. ML module makes an effective expansion on data to achieve a balance between data shortage and redundancy. It especially avoids the uneven distribution of data features which is essential in the analysis.

[Editor's note: Please replace the "metadata" with another word.]

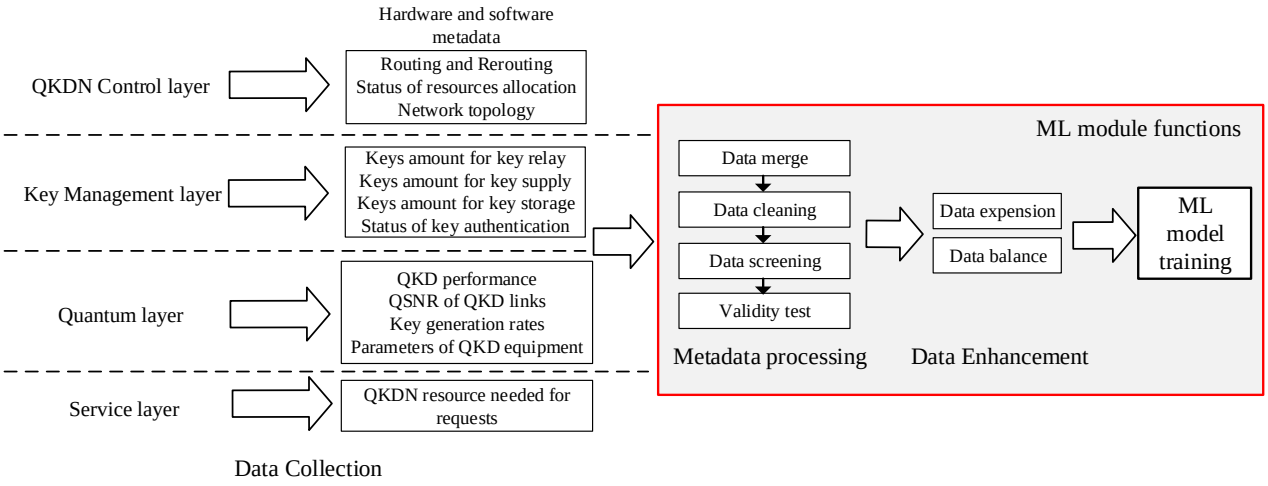


Fig. 9.1 Diagram of ML-based data collection and pre-processing

Use case analysis

- Analysis related to data collection

  1) It needs to collect the static data from each layer (e.g., the QKDN hardware and software metadata, metadata about physical and virtual resources [such as the network topology, resource usage data and QKDN equipment data]).

  2) It needs to collect the dynamic data from each layer (e.g., the performance data, configuration status, inventory and life cycle of the QKDN resources).

  3) It needs to collect the event logs of each layer.

- Analysis related to data processing

1) It needs to support storage of collected data.
2) It needs to support the transformation of data into expected format (e.g., the categorizing and aggregating of multi-sourced data).

- Analysis related to application of ML output

None.

Benefits and impact

The ML-based data collection is to collect multi-source, heterogeneous QKDN metadata and transformed it into understandable, unified and easy-to-use structures for analysis.

9.3. Use case CML02: ML-based routing

Use case title

ML-based routing

Use case description

(1) Background

When the service comes, the appropriate forwarding path needs to be selected according to the key required by the service. Now, the construction of a series of small-scale QKD technology verification networks has been completed. However, the key generation rate in the current QKD network is relatively low, which cannot meet the needs of data encryption. QKP exists between two adjacent QKD nodes and is used to achieve key resource management. When there is a key requirement between the communication nodes, the control and management layer select a route to satisfy the requirement with the QKP supplying keys.

(2) Issue

Due to the dynamic and explosive nature of services, the generation and consumption of critical resources are often unbalanced. When the number of keys on the chosen path does not meet the requirements of service encryption, the success rate of subsequent services is reduced. Besides, performing optimal routing of several end-to-end connections is typically a very complex task. Thus, efficient algorithms, such as those enabled by machine learning (ML), are needed to provide accurate network reconfiguration in reasonable time.

(3) Role of ML in QKDN

[Editor's note: Please add the routing related description in Y.3803.]

The control and management layer is able to acquire network information, such as network topology, resources' updates, real-time bandwidth requests, link load, key consumption rate, residual amount of keys, etc. And we use them to implement optimization algorithms. Given this large number of information available from the control plane, it is possible to train machine learning algorithms to automatically and continuously optimize the network.

The role ML plays in routing is shown in Fig 9.2. Firstly, it is necessary to obtain various information about status of links and keys. In the next module the machine learning model is trained. From a view of machine learning, we thought of routing as a classification problem. Machine learning, such as the classification module, reinforcement learning, is actually used to classify the routing parameters. The output of this sub-module is the optimal routing scheme which is passed to the QKDN control layer. Finally, the routing scheme obtained by the classifier is appropriately translated into rules of flow for network to accomplish the optimal routing.

Furthermore, the machine learning module can obtain the routing configuration and real-time network re-configuration.

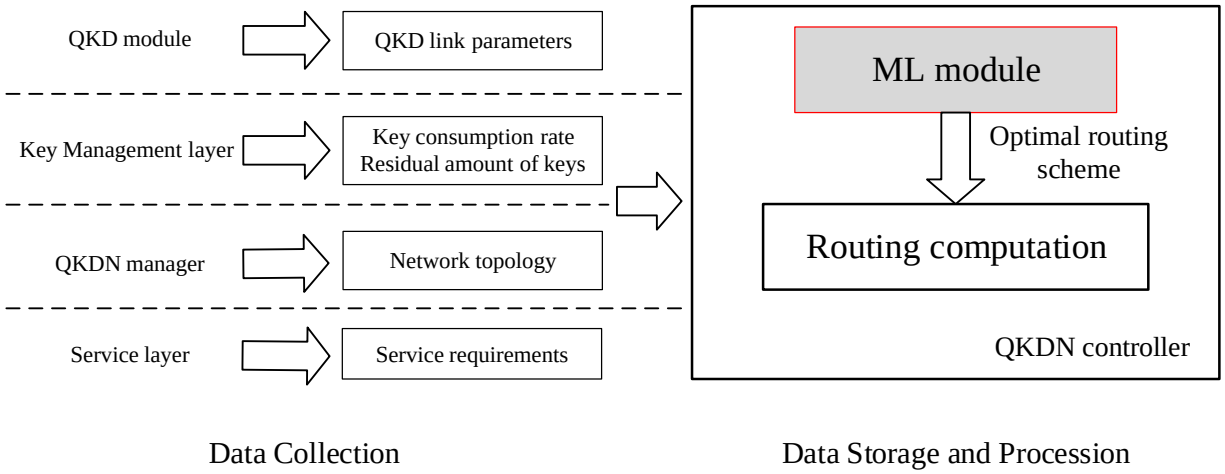


Fig. 9.2 Diagram of ML-based routing

Use case analysis

- Analysis related to data collection

  1) It collects information on key consumption rate and residual number of keys from the KM layer.
  2) It collects QKD link parameters from the QKD modules and QKDN topology information from QKDN manager.

- Analysis related to data storage and processing

  1) It manages a routing table which contains necessary information on QKD node addresses and KM IDs.
  2) It supports an optimization of key relay routes in the entire QKDN by the QKDN manager, which monitors the whole status of the quantum layer and the key management layer, and registers and updates them in a database.

- Analysis related to application of ML output

  1) The ML output is applied in assessing the key resources of the QKPs.
  2) The ML output is applied to select an optimal routing strategy.

Benefits and impact

The ML-based Resource Allocation can effectively reduce the number of times the key resources exceed the expected range, thereby reducing the service blocking rate, improving the key resources utilization and ensuring the security performance of the key.

[Editor's note: Please consider different scenarios such as real-time, near real-time, non-real-time.

9.4. Use case CML03: ML-based QKDN fault diagnosis

Use case title

ML-based QKDN fault diagnosis

Use case description

(1) Background

The key control management detects and collects the alarm information of the key control layer, key management layer and quantum layer, monitors performance of QKDN layer's functions, which is stated in the recommendation ITU-T Y.3804 "Quantum key distribution networks - Control and management". In order to avoid network faults more effectively, the QKDN control management

should be able to locate and predict faults in time according to the monitoring data and alarm information of each layer.

(2) Issue

In the process of service transmission, QKDN fails and both sides of communication cannot generate the key through negotiation. If the key resources in QKP are not enough to support the encryption of the communication data between two parts, then eavesdropping attacks may lead to communication insecurity in a short period of time. In this case, network fault recovery and service reconstruction will introduce time cost and high risk of data breach. The key control management controls, monitors and manages the entire QKDN. As recommended by the Req_M 1 and Req_C 7 in [ITU-T Y.3801], the QKDN controller provides fault information to a QKDN manager and the QKDN manager analyzes the status information collected/received for fault indicators. Based on the fault information in QKDN, the key control management is able to use ML, which is an automatic and efficient data analysis tool to realize QKDN fault diagnosis and business reconstruction to ensure the security of communication.

(3) Role of ML in QKDN

The ML-based QKDN fault diagnosis solution locates and predicts the fault in QKDN before the network fault which is shown in Fig. 9.3. ML model collects the alarm information of each layer in the QKD network and the training model of a large amount of data in the current network operation. Through the fault classification induction and feature analysis, the ML model is aware of the association rules and related information between the alarm information and the fault in QKDN. In combination with the alert knowledge base and the related information about the network topology, it can use the automatically extracted feature representation to perceive the complex correlation between multiple faults and diagnose the faults.

Convolutional neural network has few parameters but powerful feature extraction and representation ability. Full-connected network has simple structure, it can fully extract potential valuable information from data. It is suggested that these two models can be used to build the mathematical model of fault diagnosis.

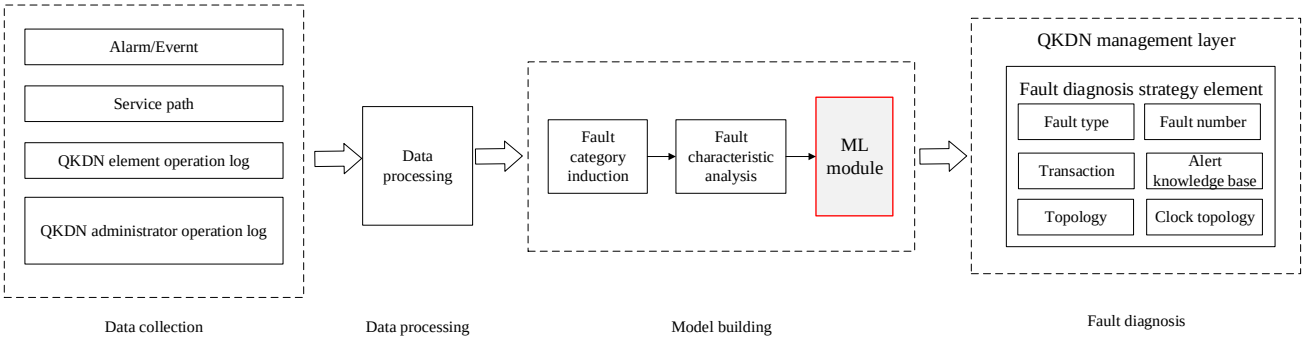


Fig. 9.3 Diagram of ML-based QKDN fault diagnosis

<u>Use case analysis</u>

- Analysis related to data collection

  1) It collects the alarm information of the key control layer, key management layer and quantum layer.
  2) It collects the data on the operation of the current QKDN.

- Analysis related to data storage and processing

  1) It supports storage of data used for analytics.
  2) It supports retraining the ML model after the update of datasets.

- Analysis related to application of ML output

    1) The ML output is applied in supporting survivability protection in advance in the network fault.
    2) The ML output is applied in service protection in QKDN.

Benefits and impact

The ML-based QKDN fault diagnosis solution will reduce the time cost and the risk of network fault by carrying out fault location and fault prediction based on the alarm information.

Appendix I – Gap analysis

ETSI GS QKD 002 gives the use cases of QKD including offsite backup/business continuity, enterprise metropolitan area network, backbone protection and so on.

ITU-T FG QIT4N describes the use cases of QKDN in vertical and horizontal domains.

IETF/IRTF Quantum Internet Research Group (qirg) has studied the applications and use cases for the quantum Internet.

ITU T Y-series Recommendations-Supplement 55 describes use cases of machine learning in future networks including IMT 2020.

The living list "Machine learning in quantum key distribution networks" was first proposed in Q16/13. It studies the role of ML in QKDN on the perspective of networking and involves several use cases of ML application in QKDN including routing selection, fault diagnosis, and so on.

Then, the proposal of adding uses cases of machine learning based quality of service (QoS) assurance for QKDN was proposed in Q6/13. But the use cases in Draft Recommendation Y.QKDN-qos-ml-req only limits to the ML application on the aspect of QoS assurance requirement in QKDN.

Thus, it is necessary to further study the applications of ML in QKDN comprehensively as a supplement .

Bibliography

Ou, Y., et al., "Field-Trial of Machine Learning-Assisted Quantum Key Distribution (QKD) Networking with SDN". 2018 European Conference on Optical Communication (ECOC), 2018.

K. Abdelli, H. Grießer and S. Pachnicke, "Machine Learning Based Data Driven Diagnostic and Prognostic Approach for Laser Reliability Enhancement," 2020 22nd International Conference on Transparent Optical Networks (ICTON), Bari, Italy, 2020, pp. 1-4.

[ETSI GS QKD 002] ETSI GS QKD 002 (2010), Quantum Key Distribution; Use Cases.

[IETF/IRTF QIRG] IETF/IRTF QIRG (2020), draft-irtf-qirg-quantum-internet-use-cases-03, Applications and Use Cases for the Quantum Internet.

[ITU-T SG13-LS154] LS on the living list (2020), Machine learning in quantum key distribution networks.

[ITU-T T17-SG13 Temporary Document 671-WP1] Draft new Recommendation ITU-T Y.QKDN-qos-ml-req (2020), Requirements of machine learning based QoS assurance for quantum key distribution networks.

_____