Towards Optimized Demand Routing in QKD Networks

Mario Wenning[1,2], Sai Kireet Patri[1,2], Jasper Müller[1,2], Achim Autenrieth[1], Jörg-Peter Elbers[1], Piotr Rydlichowski[3], Carmen Mas-Machuca[2]

[1]ADVA, Fraunhoferstrasse 9a, Munich, Germany [2] Chair of Communication Networks, Technical University of Munich, Munich, Germany [3] PSNC, Poznań, Poland

mwenning@adva.com

Abstract: We investigate buffer-aware demand-routing of key-consumption demands in QKD networks and implement a measurement-based framework for path selection. The proposed ML-based algorithm outperforms state-of-the-art heuristics by 22 % and 92 % for networks under test. © 2022 The Author(s)

1. Introduction

The requirement of secure data communications is continuously becoming more strict and demanding. Additionally, the threat of quantum computers challenges classical cryptography to protect data. One promising solution is quantum key distribution (QKD). QKD hardens the key exchange against quantum computer attacks by facilitating sharing of a secret key generated by using a quantum channel between both parties [1]. The quantum properties, e.g., the No Cloning Theorem [2], ensure the privacy of the secret key. It is impossible to perform meaningful observations of the quantum channel without being noticed by the legitimate transmitter and/or receiver [3].

We conducted measurements of QKD devices, deployed in the field, that show significant fluctuations in the secure key rate (SKR). The SKR is the number of bits secretly shared between the transmitter and receiver within a time frame and characterizes the performance of the quantum channel. Key-buffers at every node store the keys until usage. For the maximum utilization of QKD in differently-sized optical network topologies, the routing of key-consumption demands needs to be optimized to avoid overfilling and draining of key-buffers. We define this issue as the buffer-aware demand-routing problem (BADRP). Solving BADRP allows the operator to maximize the key size with a given refresh rate and to cope with fluctuations in the key generation rates.

Our contribution is twofold. Firstly, we dimension the QKD network based on measurements. Secondly, we define the BADRP and implement three different algorithms, based on heuristic, integer linear programming (ILP), and machine learning (ML), for routing demands in the QKD network. Finally, we compare their performances at the maximum possible key size. Our results show that the proposed ML-based routing algorithm allows operation with the maximum key size on a demand-by-demand basis while achieving maximum utilization of the studied networks.

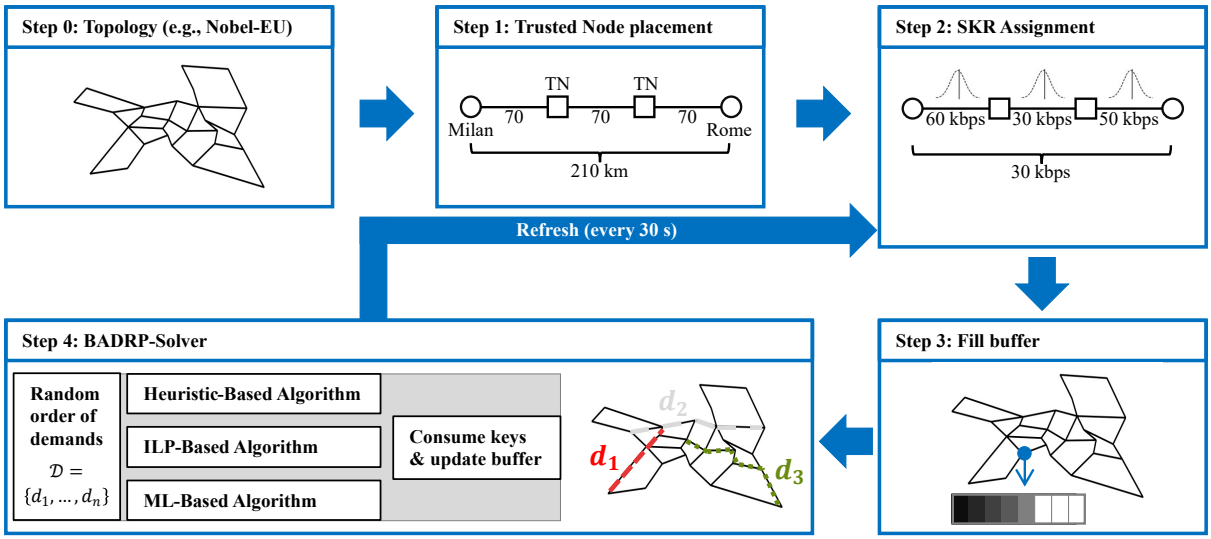
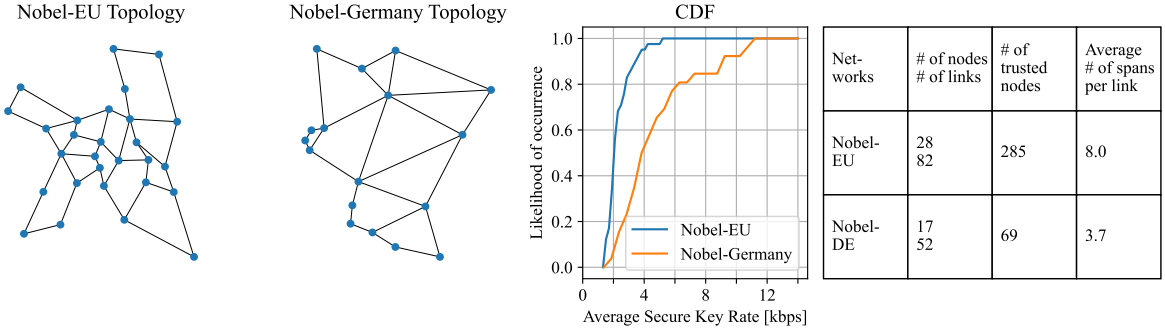


Fig. 1. Schematic summary of the QKD network simulation framework and the comparison of the routing algorithms.

2. Proposed framework and routing algorithms

As depicted in Fig. 1, our framework consists of five steps to define the QKD network, simulate the SKRs, and evaluate the algorithms. Firstly, we define the topology with bidirectional links (Step 0). In Step 1, we perform the placement of trusted nodes (TNs). We consider the influence of the length of the quantum link as a piece-wise approximation. For links up to 80 km, the framework takes fiber attenuation into account. For longer links, TNs are required. We split the link into spans such that the longest span does not exceed the threshold (80 km). The

framework optimally places TNs and divides the link into spans of equal length, while maximizing the SKR. We measured the SKR from commercial QKD systems (idQuantique Cerberis [4]) deployed in a live network over several months. These QKD systems are operated over five fiber spans with different lengths between Warsaw and Poznan. Measurements show fluctuations over time with a standard deviation of approx. 9 % of the mean. Most of the spans show clear degradation of the SKR due to span length and allow for generalization. The measurement series is the basis for our QKD network simulation framework. We approximate the SKR with the idealized rate based on the BB84-protocol [5, 6]. We model the SKR with a Gaussian distribution to simulate the fluctuations and assume a stable SKR for 30 seconds. The SKR of the link is bounded by the lowest SKR of its spans (Step 2). Fig. 2 shows the topologies and the cumulative distribution function (CDF) of the resulting average SKR for the analyzed networks [7]. Furthermore, it states the number of TNs required to achieve the CDF. We assume an equally sized buffer at every node for every link. The number of bits filling the buffer every 30 seconds is sampled from Gaussian distributions (Step 3). We treat the key consumption as a demand between two nodes. The demands reflect the request for secure communication between any pair of nodes, and the refresh rate for the used key is equal to 30 seconds. Due to static encrypting hardware, the demands are constant. We randomly change the order of the demands at every refresh instance to relax the requirement on synchronization. For each demand, the BADRP-solver chooses one path from hop-based k-shortest paths by avoiding a key shortage in consecutive loops. The BADRP-solver's objective is to maximize the least filled buffer and avoid overfilling. The optimal path depends on the status of the buffers, the SKRs along the paths, and the routing of the previous demands. Based on the selected paths, the key consumption is computed and the buffers are updated accordingly for the next refresh (Step 4). To the best of our knowledge, there are no works that provide heuristics to solve the BADRP in long-haul



Net-works	# of nodes # of links	# of trusted nodes	Average # of spans per link
Nobel-EU	28 82	285	8.0
Nobel-DE	17 52	69	3.7

Fig. 2. Analyzed networks with the corresponding CDF of the average secure key rate.

QKD networks. However, heuristics for routing demands based on congestion management have been applied to other networks, such as Time Sensitive Networks (TSN). Since the problem definition is mathematically similar, we implement a threshold-based routing heuristic [8], which serves as a baseline. To compete with the baseline, we contribute two algorithms, the first based on ILP and the second based on ML. The ILP-based algorithm optimally places all demands simultaneously at every refresh instance. The ILP-based algorithm is treated as an upper bound for performance. In contrast, our second proposed algorithm based on ML works on a demand-by-demand basis. We train the classifier based on the XGBoost framework using the softmax objective [9]. During the learning phase, the classifier uses the ILP-based decision as a ground truth. The number of hops, least filled buffer, shortest-path betweenness centrality for the nodes, and average SKRs are the inputs of the classifier. With this choice of parameters, the training time is in the range of several minutes. The comparison between the ML- and the heuristic-based algorithm is consistent because both algorithms work on a demand-by-demand basis and do not share information about future demands. Since the learning depends on the network's topology, we test the Nobel-EU and the Nobel-Germany (Nobel-DE) networks [7].

3. Comparison of the three routing algorithms

Larger key sizes between endpoints allow for more encrypted traffic sent via the network. Therefore, we first analyze the maximum possible key size for homogeneous demands between any pair of nodes using the three different algorithms. Starting with the heuristic, we study the influence of the congestion threshold and the weight parameter K of the cost function for maximizing the key size [8]. We observed the best performance with a congestion threshold of 130 demands for the Nobel-EU and 47 demands for the Nobel-DE networks. The congestion recovery reduces the influence of K. For the heuristic, $K = 1$ in combination with the congestion thresholds achieves the maximum key sizes of 0.795 and 1.807 kbit for the Nobel-EU and Nobel-DE networks, respectively. Applying the refresh rate of 30 seconds, the key sizes correspond to the consumption of SKRs of approx. 26 and 60 bps. Both proposed algorithms equally increase the key size by approx. 22 % for the Nobel-EU and by approx. 92 % for the Nobel-DE networks as compared to the heuristic. The increase leads to 0.971 and 3.462 kbit key sizes for the Nobel-EU and Nobel-DE networks, respectively. Furthermore, we analyze the states of the buffers for evaluating

the algorithms. The results show 480 cycles corresponding to 4 hours with a refresh every 30 seconds. We allow up to 5 shortest paths in the networks and set the key sizes to 0.971 and 3.462 kbit for all routing algorithms. The buffer size is equal to 400 kbit. Fig. 3 depicts the results. On the left side, Fig. 3 focuses on the least filled buffer within the network depending on the number of cycles. On the right side, Fig. 3 shows boxplots that summarize the buffers' states for the different routing algorithms. The ILP-based algorithm achieves stability of the least filled buffer by at least 44 % for both networks. The fluctuations using the Nobel-EU network are large compared to the Nobel-DE network due to an increased number of spans per link. Considering the ML-based algorithm and the Nobel-EU network, the least filled buffer drops at the beginning to approx. 47 % but maintains a stable level throughout the simulation. The observation also holds for the Nobel-DE network with a performance comparable to the ILP-based algorithm. The ML-based algorithm achieves stability of the least filled buffer by at least 40 % for both networks. The heuristic shows a steep drop within the first cycles for both networks. This deficiency affects at least one edge and is evident in the boxplots in Fig. 3. Most of the buffers are filled over 50 %, but a few edges are the bottleneck and drain quickly. In contrast, ML-based routing distributes the load of the demands more efficiently. One major problem of the heuristic is the identification of a congested edge. It is hard to find a threshold such that enough edges are relieved, while having sufficient remaining edges for demand distribution. Comparing computational time, the ML-based algorithm routes each demand within 5 ms independent of the network size. The computational time for the heuristic increases with the network size. It takes 3 ms for the Nobel-DE and 7 ms for the Nobel-EU network to place one demand on an average. Since the ILP-based algorithm places the demands simultaneously, it takes advantage of parallelizing and averages at less than 1 ms per demand.

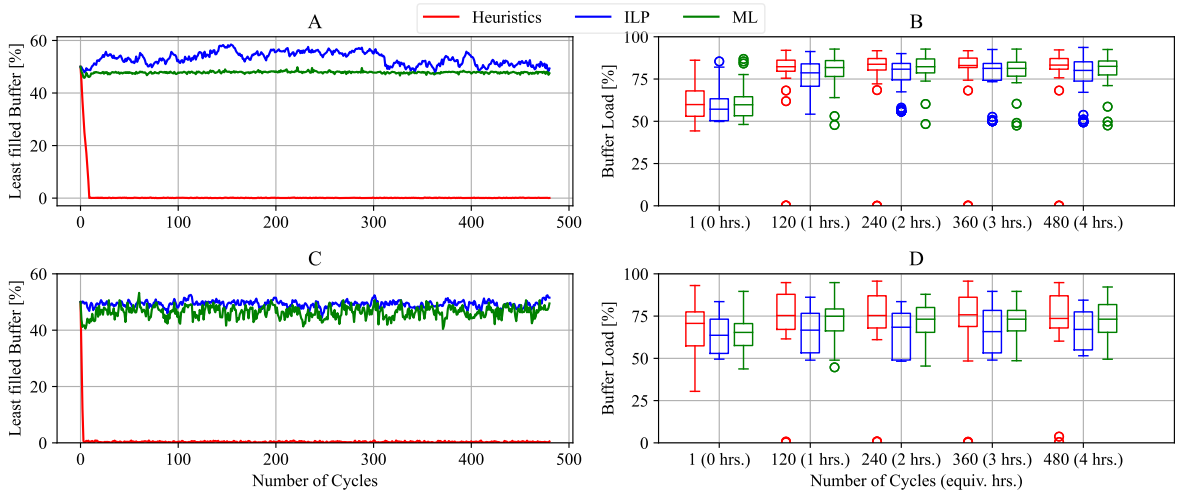


Fig. 3. A and C depict the evolution of the least filled buffer during operation for the Nobel-EU and Nobel-DE networks, respectively. B and D show the states of all buffers for the Nobel-EU and Nobel-DE networks, respectively.

4. Conclusion

We have analyzed the optimal routing problem for key-consumption demands in a QKD network simulation framework inferred from a measurement series of a live network. We proposed two algorithms and compared them to the state-of-the-art heuristic. The ILP-based algorithm performs optimally if all demands are known. Using the ML-based algorithm, the operator can also utilize the network to the limit and route on a demand-by-demand basis. The ML-based algorithm enables operation at the maximum key size without overfilling or draining individual buffers. Furthermore, it ensures operation with limited demand information and increases resiliency against dynamic demand changes. Compared to the heuristic, the performance is increased significantly. One drawback of our proposed solution is the topology dependency. However, the heuristic also depends on the topology and has to be manually adapted. The manual adaption is automated for the ML-based algorithm.

The work has been partially funded by the German Ministry of Education and Research in the project QuNET+ML (**#16KISQ066**).

References

1. V. Scarani *et al.*, "The security of practical quantum key distribution," Rev. Mod. Phys., pp. 1301–1350, 2009.
2. S. Imre *et al.*, "Quantum Computing Basics", In Quantum Computing and Communications, ch. 2, 2004.
3. H.-K. Lo *et al.*, "Unconditional Security of Quantum Key Distribution over Arbitrarily...", Science, **283**, (5410), 1999.
4. ID Quantique, "ID Quantique", [Online], Available: http://www.idQuantique.com/, [Accessed: 22-Aug-2022].
5. E. Diamanti, "Security and implementation of differential phase shift quantum key distribution systems", 2006
6. C. H. Bennett *et al.*, "Quantum cryptography: Public key distribution and coin tossing", arXiv:2003.06557, 2020.
7. "SNDlib", Zuse-Institute Berlin, [Online], Available: https://sndlib.zib.de, [Accessed: 22-Aug-2022].
8. M. A. Ojewale *et al.*, "Routing heuristics for load-balanced transmission in TSN-based...", SIGBED, **16**(4), 2020.
9. T. Chen *et al.*, "XGBoost: A Scalable Tree Boosting System", Proc. ACM SIGKDD, pp. 785–794, 2016.