

Spatio-Temporal Buffer Management for Relayed QKD Networks

Konstantinos Christodoulopoulos, Antonis Selentis, George T. Kanellos

Department of Informatics and Telecommunications, National and Kapodistrian University of Athens, Greece

e-mail: kchristodou@di.uoa.gr

ABSTRACT

We address dynamic key management in Relayed Quantum Key Distribution (QKD) networks under fluctuating Secret Key Rate (SKR) generation and demand. We propose a periodic batch-based scheme that maintains per-pair Virtual Quantum Key Pools (VQKP) at the level needed for service while letting per-link Quantum Key Pools (QKP) absorb generation fluctuations across periods. The relay allocation is formulated as a path-based multicommodity-flow LP over precomputed multi-path sets. On a uniform 15-node ring with realistic fluctuations, the scheme stabilizes worst-pair key delivery and at least doubles availability compared to fully proactive and reactive baselines.

Keywords: Quantum Key Distribution, key relay, dynamic key management, multicommodity flow, linear programming, buffer management.

1. INTRODUCTION

Quantum Key Distribution (QKD) leverages quantum-mechanical principles to securely distribute symmetric keys between two endpoints, and is the most mature quantum technology with commercial systems already deployed [1]. Extending QKD beyond point-to-point links to network-wide many-to-many key distribution remains an open challenge [2][3]. *Switched QKD*, where optical circuit switches are used to dynamically establish direct QKD links between nodes is one solution. The most widely deployed network architecture is *Relayed QKD*, where adjacent QKD nodes generate keys directly while non-adjacent nodes communicate via trusted intermediate nodes that forward keys using one-time-pad (OTP) encryption.

In a deployed QKD network, both the SKR generated on each link and the key demand from each pair of Secure Application Entities (SAEs) fluctuate, and neither is known ahead of the time. The Key Management System (KMS) layer must allocate relay capacity in this uncertain regime, and the choice of allocation policy determines whether the network meets its service-level agreement (SLA) under operating conditions.

We can classify allocation schemes among two axes. Based on *provisioning timing*, schemes are either reactive [4], relaying keys upon request, or proactive [5], pre-filling Virtual Quantum Key Pools (VQKPs) so requests are served immediately. Reactive schemes suffer from serving delays, while proactive schemes transfer keys without considering actual demands, which can lead to inefficiencies. Based on *request processing*, schemes either serve demands as they arrive, or in batches, operating in periods. Sequential serving schemes use greedy strategies that may fail at heavy loads, while batch processing may miss dynamic traffic adaptation. Beyond key-plane management, recent work has addressed data-plane scheduling for QKD-secured traffic via backpressure-style policies [6]. The methodological foundations for analysing key management in QKD networks have been developed in our prior work, including analytical formulas for uniform rings [2] and ILP-based formulations for non-uniform rings [3].

To keep the best attributes of prior works, we propose a “just-enough” proactive approach that absorbs SKR generation fluctuations, adapts to dynamic network traffic, and periodically calculates global optima, avoiding path overlap and resulting in maximized key consumption. Furthermore, our proposed scheme uses multipath routing to enhance efficiency and resilience, which can be extended to increase security [7],[8]. So, in this work, we propose a periodic batch-based dynamic key management scheme for Relayed QKD networks under fluctuating SKR generation and demand. Our contributions are as follows:

1. A *spatio-temporal formulation* of dynamic key management based on a two-level pool model: per-link Quantum Key Pools (QKPs) absorb generation fluctuations across periods, while per-pair VQKPs serve demands. This couples spatial flow allocation with temporal buffer evolution and shifts the problem from spatial load balancing to spatio-temporal buffer management.
2. A *multi-commodity Linear Programming (LP) formulation* to balance the two-level pool that supports multiple parallel paths per pair, which enables load balancing across the network and enables security mechanisms against intermediate-node compromise (e.g., secret sharing across paths).

We evaluate the proposed scheme on a uniform 15-node ring under fluctuating SKR and demand, comparing against batch-based fully proactive and a sequential reactive baselines.

2. NETWORK ARCHITECTURE AND KEY DYNAMICS

We assume that the QKD network consists of three layers, as shown in Fig. 1. Layer 1 is the *Quantum Layer*, where adjacent QKD nodes use Alice–Bob device pairs to generate symmetric keys at the secret key rate (SKR). Layer 3 is the *Service Layer*, where Secure Application Entity (SAE) pairs at different nodes consume symmetric keys to secure their communication. Layer 2 is the *Key Management System (KMS)* layer. Each node hosts a KMS server, which stores the keys generated by its adjacent QKD links in *Quantum Key Pools (QKPs)*—one QKP per QKD link.

For non-adjacent SAE pairs, keys are *relayed* through intermediate trusted nodes. Each intermediate node along a relay path uses one-time-pad (OTP) encryption: it consumes one local key from the QKP of the previous link and one from the QKP of the next link, forwarding the relayed key without ever exposing it. The resulting end-to-end relayed

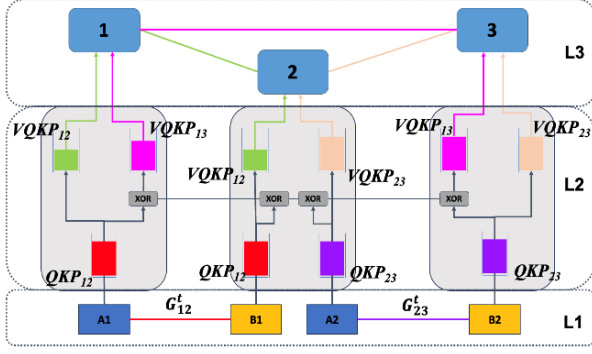


Fig. 1: Typical 3-layered QKD network architecture, focusing on KMS Layer.

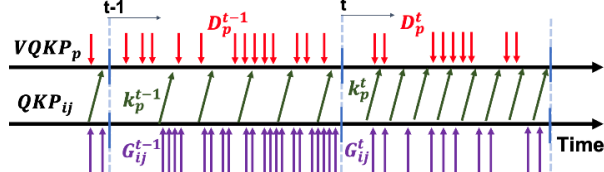


Fig. 2: Schematic for the evolution of buffer pools over time. Keys arrive at varying rate G , and stored at QKP. Some of them are used to create relay keys (e.g. with constant rate decided by the algorithm) that are stored at VQKPs. Request arrive randomly over time.

consider a period successful if $C_{sd}^t = D_{sd}^t$ for all $a_{sd} \in A$. Let $g_{sd,k}^t$ denote the keys delivered to $VQKP_{sd}$ via path $p_{sd,k}$ in period t ; each link l_{ij} along $p_{sd,k}$ contributes $g_{sd,k}^t$ local keys from its QKP_{ij} to enable this delivery. The total keys delivered to $VQKP_{sd}$ in period t , summed across all paths used, is $g_{sd}^t = \sum_k g_{sd,k}^t$.

The pool sizes evolve from period t to period $t + 1$ as:

$$QKP_{ij}^{t+1} = QKP_{ij}^t - \sum_{a_{sd} \in A} \sum_{p_{sd,k} \in P_{sd}, l_{ij} \in p_{sd,k}} g_{sd,k}^t + G_{ij}^t \quad (1)$$

$$VQKP_{sd}^{t+1} = VQKP_{sd}^t - C_{sd}^t + g_{sd}^t \quad (2)$$

QKP_{ij} grows by what was generated on the link in period t and shrinks by the local key consumption induced by every relay path crossing the link. $VQKP_{sd}$ grows by the keys delivered to the pair $s-d$ across all its (precomputed) paths and shrinks by what the SAE consumed. The path-based variable $g_{sd,k}^t$ cleanly captures the multi-hop accounting: each path consumes the same volume on every link it traverses. Fig. 2 schematizes this evolution.

The pool sizes at the start of period $t + 1$ are known once period t has ended, but the keys that will be generated and demanded within period $t + 1$ are not. This sets up the optimization problem treated next: how to allocate the path flows $g_{sd,k}^t$ so that VQKPs remain populated for the SAEs while QKPs absorb the SKR fluctuations.

3. PROBLEM AND PROPOSED SCHEME

As discussed, proactive schemes maximize per-period transfers from QKPs to VQKPs without considering current demand; over time, this disproportionately benefits short paths that accumulate fluctuations over fewer links, leaving long-path VQKPs underserved. Reactive schemes avoid this imbalance but suffer the serving delay of an on-demand relay computation and rely on greedy per-request decisions that cannot smooth fluctuations. We propose a just enough proactive scheme that balances the levels of QKPs and VQKPs. At each period, we solve a global allocation to deliver enough relayed keys to maintain VQKPs, by setting the target amount of relayed keys equal to the cumulative consumption of the previous period, while allowing the QKPs to absorb generation fluctuations. So, the scheme operates in periods of duration T . At the start of period $t + 1$, the demand D_{sd}^{t+1} that will arrive within the period is unknown; we therefore cannot use it directly as a relay target. Under traffic that is roughly stable across consecutive periods — that is, when D_{sd}^{t+1} is close to D_{sd}^t — the previous period's consumption is a usable proxy. We set the per-pair relay target as $K_{sd}^{t+1} = C_{sd}^t$. Under highly bursty traffic this lag of one period is a real cost, and it is one of the open issues addressed in our extension work outlined in Section 5.

We formulate the per-period allocation as a multicommodity-flow Linear Programming (LP) problem that determines path flows $g_{sd,k}^{t+1}$ delivering exactly the targets K_{sd}^{t+1} while spreading link consumption across the available QKPs. Lowercase symbols denote variables and uppercase denote constants.

Constants.

- QKP_{ij}^{t+1} : size of QKP for link l_{ij} at the start of period $t + 1$.
- K_{sd}^{t+1} : target keys to be delivered to $VQKP_{sd}$ in period $t + 1$.
- $\text{MaxQKP}^{t+1} = \max_{l_{ij} \in L} QKP_{ij}^{t+1}$.
- ϵ : small positive constant for numerical stability.

key is stored in a *Virtual Quantum Key Pool* (VQKP) at the source and destination end-nodes, from which the SAEs draw. For adjacent SAE pairs, VQKPs are filled directly from the QKPs without relaying. The two pool levels play complementary roles: QKPs are *generation-side* buffers that absorb SKR fluctuations, while VQKPs are *service-side* buffers that decouple SAE consumption from the relay computation. Multiple paths can be used in parallel for the same SAE pair, increasing survivability under link or node failures and enabling load balancing.

We consider the dynamic evolution of the network over time and re-adapt it at a constant period T . We denote by N the set of nodes, by L the set of QKD links and by A the set of SAE pairs, where each pair $a_{sd} \in A$ consists of a source node s and a destination node d . For each pair a_{sd} , we precompute a set of relay paths P_{sd} , with $p_{sd,k} \in P_{sd}$ denoting the k -th path. For period t , we denote by G_{ij}^t the keys generated on link $l_{ij} \in L$ between adjacent nodes (i, j) , and by D_{sd}^t the sum of demands for pair a_{sd} that arrived within that period. Both G_{ij}^t and D_{sd}^t are unknown at the start of period t . We denote by C_{sd}^t the keys consumed for pair a_{sd} within period t , and

Variables.

- $g_{sd,k}^{t+1} \geq 0$: keys delivered to VQKP_{sd} via path $p_{sd,k}$ in period $t + 1$.
- w_{sd}^{t+1} : weighted relay cost for pair a_{sd} .

Objective.

$$\text{minimize } \sum_{a_{sd} \in A} w_{sd}^{t+1} \quad (3)$$

Constraints.

C1 (path delivery):

$$\forall a_{sd} \in A, \sum_{p_{sd,k} \in P_{sd}} g_{sd,k}^{t+1} = K_{sd}^{t+1} \quad (4)$$

C2 (link capacity):

$$\forall l_{ij} \in L, \sum_{a_{sd} \in A} \sum_{p_{sd,k} \in P_{sd}, l_{ij} \in p_{sd,k}} g_{sd,k}^{t+1} \leq \text{QKP}_{ij}^{t+1} \quad (5)$$

C3 (cost weighting):

$$\forall a_{sd} \in A, w_{sd}^{t+1} = \sum_{p_{sd,k} \in P_{sd}} g_{sd,k}^{t+1} \cdot \sum_{l_{ij} \in p_{sd,k}} \frac{\text{MaxQKP}_{ij}^{t+1}}{\text{QKP}_{ij}^{t+1} + \varepsilon} \quad (6)$$

The objective in (3) minimizes the total weighted relay cost. C1 ensures each pair receives exactly its target K_{sd}^{t+1} . C2 caps total local key consumption per link by the available QKP; the inner sum reflects that every relay path crossing l_{ij} consumes its full path flow $g_{sd,k}^{t+1}$ from QKP_{ij} — once per link traversed, not twice, since QKP buffers are symmetric. C3 weights each path by the inverse of its links' QKP sizes, normalized by MaxQKP_{ij}^{t+1} , steering allocation toward links with more keys.

If the LP is infeasible, we reduce the targets proportionally and re-solve. The LP is a continuous multicommodity-flow problem over a precomputed path set and is solvable in polynomial time. The scheme has been implemented in a field-deployed multi-vendor 5-node QKD ring (paper under review), where the LP runs well under one second per period. This formulation builds on prior multicommodity-flow approaches to key relay [5] but differs in the two-pool problem structure that lets QKPs absorb SKR fluctuations while VQKPs serve demand, and in the targeted relay $K_{sd}^{t+1} = C_{sd}^t$ that maintains VQKPs at exactly the level needed for service. This shifts the problem from spatial flow allocation to spatio-temporal buffer management.

4. NUMERICAL EVALUATION

We evaluated the proposed scheme on a uniform ring of $N = 15$ nodes, with adjacent QKD links of 10 km and 0.2 dB/km attenuation. The per-link SKR was sampled around a mean derived from the Toshiba QKD devices in our laboratory [9], with parametric standard deviation $\sigma \in \{0, 0.1, 0.2\}$. SAE demands were uniform across all node pairs and expressed as a fraction of the maximum sustainable per-pair rate $D_{\max}$, computed from the analytical formulas for uniform rings derived in [2]. Simulations run for 400 periods of duration T . We measured two metrics per period: the *minimum* number of keys delivered across all SAE pairs (worst-pair throughput) and the *sum* of keys delivered (aggregate throughput). A period was recorded as blocked if any pair received less than its demand.

We compared three schemes. The *proactive* scheme (variation of [5]) maximizes per-period transfers from QKPs to VQKPs without considering current demand; any leftover capacity at period end is also pushed to VQKPs. The *reactive* scheme works in quasi-real-time, it operates in subperiods of 1/10 of the normal period, serving demands in a subperiod one-by-one greedily over shortest paths. The *proposed* scheme is the periodic batch-based hybrid LP of Section 3, with $K_{sd}^{t+1} = C_{sd}^t$. Proactive and proposed run in periodic mode; reactive runs as quasi-real-time.

Fig. 3 shows the evolution of minimum and sum of keys delivered for demand $D = 0.8 \cdot D_{\max}$ (periods 1–100), $0.9 \cdot D_{\max}$ (periods 101–200), and $D_{\max}$ (periods 201–400). The proposed scheme maintained a minimum delivery near the demand throughout, including at $D_{\max}$ where the other two degraded substantially. The cause is structural: proactive's per-period maximization disproportionately fills VQKPs of short-path pairs, which absorb capacity more reliably; as a consequence long-path VQKPs are starved. The reactive scheme processes requests one at a time and cannot smooth fluctuations across pairs. Constraint C1 of our LP forces equal delivery to each pair against the target K_{sd}^{t+1} , which produces the flat minimum. The operational implication is that, at $D_{\max}$, the proposed scheme effectively eliminates blocking events while the others see substantial blocking — meaning the network can be operated closer to its theoretical capacity. The proactive scheme's *sum* of delivered keys is the highest, but this sum is concentrated at a few pairs while others are starved.

Fig. 4 shows per-period averages of minimum and sum across the three schemes for $\sigma = 0$ and $\sigma = 0.2$. At $\sigma = 0$ all three schemes performed identically — a sanity check, since without generation fluctuations the QKP buffer has nothing to absorb. At $\sigma = 0.2$, the proposed scheme achieved an average minimum 2 times higher than proactive and 3 times higher than reactive, while the proactive scheme achieved the highest sum but with the imbalance discussed above. The proposed scheme's QKPs absorbed generation fluctuations across periods, smoothing the relay process; the other two pass the fluctuations through to the SAEs. For an operator providing SLA to the worst-served pair, the

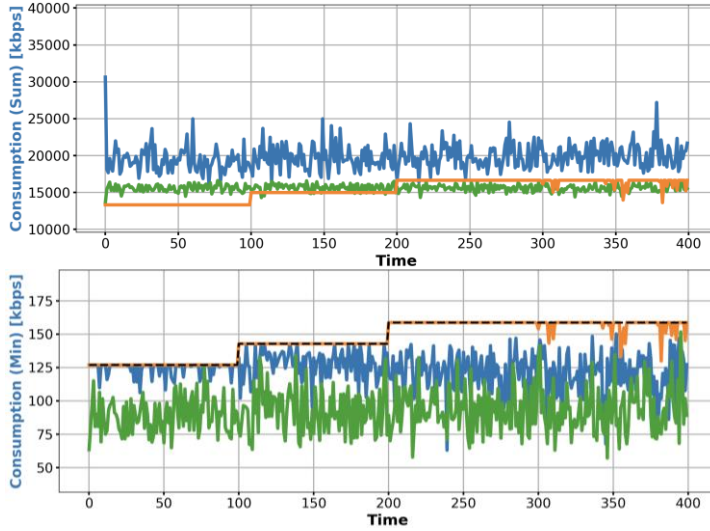


Fig. 3: Minimum and sum of keys delivered per period for the three schemes, $N=15$ ring, $\sigma=0.1$, demand levels $D=0.8, 0.9, 1.0 D_{max}$ in periods $1-100, 101-200, 201-400$.

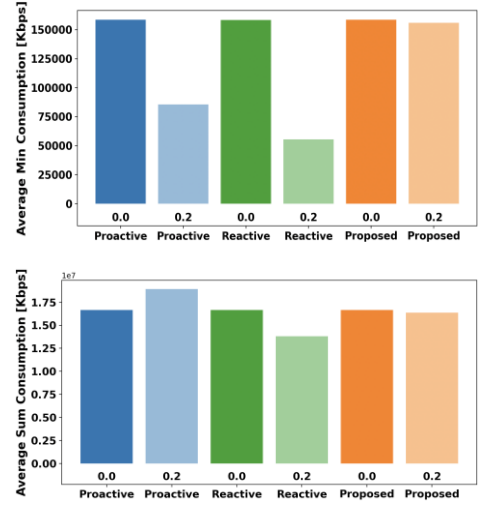


Fig. 4: Average per-period minimum (top) and sum (bottom) of keys delivered for the three schemes, $N = 15$ ring, demand $D = D_{max}$, $\sigma \in \{0, 0.2\}$.

relevant metric is the minimum: at examined fluctuation levels the proposed scheme at least doubled worst-pair availability. The fairness behaviour is visible in both figures: the proposed scheme's minimum and sum tracked each other closely, indicating evenly distributed keys across pairs, while the proactive scheme exhibited a large min/sum gap. A formal fairness metric is left to future work.

5. OUTLOOK

The formulation presented here is being extended along several directions. The $K_{sd}^{t+1} = C_{sd}^t$ target is reactive in spirit, lagging the actual demand by one period; under bursty or correlated demand, an anticipative controller-style approach is a natural improvement. Using the QKP snapshot together with the expected generation across the upcoming period lets it work with the full window rather than just its boundary. Demand heterogeneity across SAE pairs raises a fairness question that the present uniform-demand formulation does not address, and motivates a weighted variant of the LP. The journal evaluation is correspondingly broader, with time-correlated and heterogeneous demand and multiple topologies. In parallel, the scheme is being developed and validated in a field-deployed multi-vendor 5-node QKD ring testbed (paper under review). The two tracks inform each other: the testbed exposes operational issues that shape the formulation, while the formulation provides the optimization backbone the testbed runs on.

6. CONCLUSIONS

We presented a periodic batch-based dynamic key management scheme for Relayed QKD networks under fluctuating SKR generation and demand. The scheme couples per-link QKPs that absorb generation fluctuations across periods with per-pair VQKPs that serve demands and allocates relay flows via a path-based multicommodity-flow LP that targets the previous period's consumption. On a uniform 15-node ring under fluctuations, the proposed scheme stabilizes worst-pair key delivery near the demand level across the load range tested and at least doubles worst-pair availability compared to fully proactive and fully reactive baselines.

REFERENCES

- [1] Y. Cao, Y. Zhao, Q. Wang, J. Zhang, S. X. Ng, L. Hanzo, "The Evolution of Quantum Key Distribution Networks: On the Road to the Qinternet", *IEEE Communications Surveys & Tutorials*, 2022.
- [2] A. Selentis, N. Makris, A. Papageorgopoulos, P. Konteli, K. Christodouloupoulos, G. T. Kanellos, D. Syvridis, "Evaluating Relayed and Switched Quantum Key Distribution (QKD) Network Architectures", to appear in *Journal of Lightwave Technology*.
- [3] A. Selentis, K. Christodouloupoulos, G. T. Kanellos, "Relayed vs. Switched QKD: A Comparison in Non-Uniform Ring Networks", *International Conference on Optical Network Design and Modeling (ONDM)*, 2025.
- [4] Q. Zhang, O. Ayoub, A. Gatto, J. Wu, F. Musumeci, M. Tornatore, "Routing, Channel, Key-Rate, and Time-Slot Assignment for QKD in Optical Networks", *IEEE Transactions on Network and Service Management*, 2024.
- [5] C. Lee, Y. Kim, K. Shim, W. Lee, "Key-count differential-based proactive key relay algorithm for scalable quantum-secured networking", *Journal of Optical Communications and Networking*, vol. 15, no. 5, 2023.
- [6] M. S. Akhtar, K. G. Krishnakumar, V. B., A. Sinha, "Fast and Secure Routing Algorithms for Quantum Key Distribution Networks", *Transactions on Networking*, 31 (5), 2023.
- [7] M. Wang, J. Li, K. Xue, R. Li, N. Yu, Y. Li, Y. Liu, Q. Sun, J. Lu, "A segment-based multipath distribution method in partially-trusted relay quantum networks", *IEEE Communications Magazine*, 61(12), 2023.
- [8] S. M. Barnett, S. J. D. Phoenix, "Asynchronous QKD on a Relay Network", arXiv, 2012.
- [9] Toshiba QKD devices, www.global.toshiba/ww/products-solutions/securityict/qkd/products.html