다중 에이전트 강화학습 기반 완전 분산형 QKD 네트워크 제어 기법

김주봉, 이찬균, 임현교, 심규석, 이원혁
한국과학기술정보연구원

{jjbong, chankyunlee, hk.lim, kusuk007, livezone}@kisti.re.kr

A Multi-Agent Reinforcement Learning Framework in Fully Distributed QKD Networks

Ju-Bong Kim, Chankyun Lee, Hyun-Kyo Lim, Kyu-Seok Shim, Wonhyuk Lee
Korea Institute of Science and Technology Information (KISTI)

요 약

본 논문은 희소하고 소모성인 키 자원을 사용하는 QKD 네트워크에서 기존 소스 라우팅 방식의 한계점을 해결하기 위해, hop-by-hop 기반 다중 에이전트 강화학습(MARL) 라우팅 기법을 제안한다. 각 QKD 노드는 분산 부분관찰 환경에서 QMIX 알고리즘을 통해 로컬 상태에 따라 다음 홉을 동적으로 선택하며, NetSquid 기반 시뮬레이션에서 SP, WSP, HSP, HWSP 대비 성능을 평가하였다. 제안 기법은 중간 링크의 키 고갈로 인한 실패를 감소시키고 부하를 분산시켜 기존 기법보다 높은 성공률을 보였다.

Ⅰ. 서 론

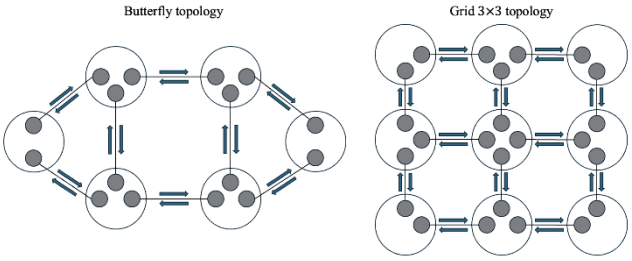
양자키분배(quantum key distribution, QKD)는 양자역학의 물리적 특성에 기반하여 이론적으로 안전한 비밀키를 생성할 수 있는 기술이다. 양자컴퓨터의 발전에 따라 기존 공개키 기반 암호체계가 붕괴될 수 있다는 우려 속에서, QKD 는 차세대 보안 인프라의 핵심 기술로 주목받는다. 최근에는 국내외에서 도시간 백본 링크를 포함한 다양한 시험망이 구축되며 QKD 네트워크(QKD network, QKDN)의 현실화가 가속화되고 있다.

그러나 QKDN 은 기존 데이터 네트워크와 달리 양자키라는 제약 조건을 지닌다. 각 링크에서 생성되는 양자키는 희소하고 소모성이며, 한 번 사용되면 복구할 수 없다. 또한 링크의 키 생성율은 수십 kbps 수준으로 낮고, 링크 길이가 길어질수록 손실과 잡음으로 인해 생성율이 급격히 감소한다. 이로 인해 링크의 키 버퍼가 고갈될 경우 해당 링크는 전송 불가 상태가 되며, 네트워크 전반의 요청 성공률이 급격히 낮아진다.

QKDN 에서는 장거리 전송 시 광섬유 감쇠와 양자 비트 오류율(quantum bit error rate, QBER)의 증가로 인해 단일 링크로 end-to-end 키를 분배하기 어렵기 때문에, 여러 중간 신뢰 노드를 경유하여 인접 노드 간에 생성된 키를 순차적으로 전달하는 릴레이 구조가 사용된다. 기존의 소스 라우팅(source routing) 기반 방식은 요청 발생 시점에 전역 최단 경로를 한 번만 계산하여 경로를 고정하는데, 경로 중간의 링크에서 키 버퍼가 고갈되면 전체 요청이 즉시 실패로 처리된다. 링크의 키 상태가 시간의 흐름에 따라 변동하는 QKDN 에서는 이러한 정적 경로 기반 접근이 높은 실패율과 낮은 자원 활용률을 초래하기 쉽다. 이러한 한계를 극복하기 위한 대안으로, 각 노드에서 로컬 상태를 기반으로 다음 홉을 동적으로 선택하는 hop-by-hop 라우팅 방식을 활용할 수 있다 [1].

본 논문에서는 각 노드를 독립적 학습 에이전트로 모델링하여 hop-by-hop 방식으로 라우팅 경로를 결정하고 요청을 스케줄링(scheduling)하는 다중 에이전트 강화학습(multi-agent reinforcement learning, MARL) 기반 제어 기법을 제안한다. 학습 알고리즘으로는 협력 기반 MARL 에서 널리 사용되는 QMIX 를 적용하며 [2], 비교 대상으로 전역 경로 기반의 shortest path (SP)와 weighted shortest path (WSP)를 선택하였다.

Ⅱ. 본 론

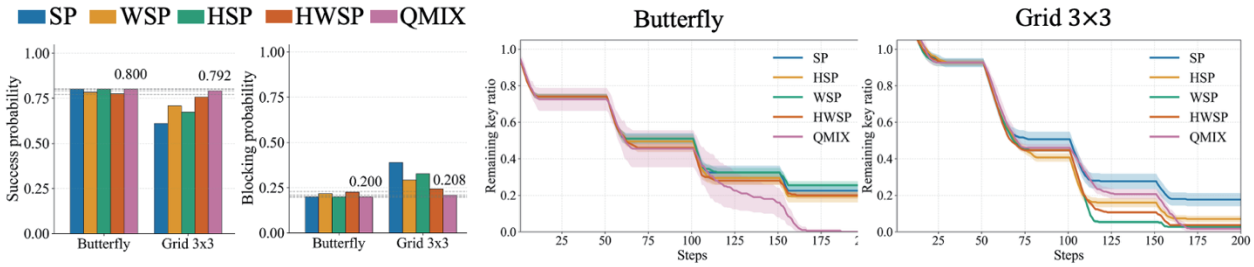


(그림 1). Hop-by-hop 기반 QKDN 에서의 Butterfly 토폴로지(좌측) 및 Grid **3×3** 토폴로지(우측).

1. 다중 에이전트 강화학습 기반 시뮬레이션 환경

실험은 NetSquid 기반으로 구현한 완전 분산형 MARL 시뮬레이션 환경에서 수행하였다 [3]. 각 노드는 BB84 프로토콜 기반의 키 생성 모듈과 키 릴레이 제어 로직을 포함하며, 강화학습 에이전트에 대응된다.

시뮬레이션은 이산 시간 기반으로 진행된다. 매 타임스텝마다 각 에이전트는 하나의 요청에 대해 거절, 인접 노드로 릴레이, 대기 중 하나의 행동을 선택한다. (그림 1)과 같이 토폴로지는 6 개 노드의 Butterfly 와 9 개 노드의 Grid **3×3**을 사용하였고, 200 스텝 동안 진행하며 50 스텝마다 가능한 모든 end-to-end 요청을 동시에 생성하였다. Butterfly 에서는 30 쌍, Grid 에서는 72 쌍의 요청이 매 50 스텝마다 생성되며 각 요청의 lifetime 은 50 스텝으로 설정하였다.



(그림 2). Butterfly 와 Grid 3×3 토폴로지에서 SP, WSP, HSP, HWSP, 그리고 QMIX 학습 완료 모델 간 성능 비교 결과. QMIX 가 기존 휴리스틱 알고리즘 대비 경로를 효율적으로 선택해 중간 노드의 키 고갈로 인한 실패율을 감소시킨 것으로 확인되며, 이는 오른쪽의 알고리즘별로 비교한 남아있는 키의 비율 그래프를 통해 검증할 수 있다.

각 링크는 제한된 양자키를 보유하며, 요청이 전달될 때마다 키가 소모된다. 키 버퍼가 고갈된 링크는 일시적으로 사용 불가능하며, 새로운 키는 에피소드 시작 시 재생성 된다. 이러한 설정은 실제 QKDN 의 키 자원 희소성과 비동기적 요청 패턴을 반영한 것이다.

2. 학습 모델과 POMDP 정의

완전 분산형 MARL 시뮬레이션 환경에서는 각 에이전트가 전역 상태를 관찰할 수 없고 자신이 위치한 노드와 인접 링크의 상태만 부분적으로 관찰할 수 있다. 따라서 전역 상태가 완전하게 관찰되지 않는 부분관찰 마르코프 결정과정(partially observable Markov decision process, POMDP)을 통해 문제가 정의된다. 에이전트의 관찰은 인접 링크의 남은 키 버퍼 비율, 현재 수신한 요청의 출발지 및 목적지와 lifetime, 그리고 큐잉 및 처리 지연(queueing and processing delay)으로 구성된다.

본 논문에서는 QMIX 를 사용하여 hop-by-hop 라우팅 정책을 학습한다. QMIX 는 각 에이전트가 개별적으로 추정한 Q 함수를 monotonic mixing network 를 통해 통합함으로써 전역 팀 보상을 극대화하는 방식으로 학습한다. 이때 각 에이전트는 전역 상태를 모두 관찰할 수 없고 자신이 위치한 노드와 인접 링크의 상태만 부분적으로 관찰할 수 있으므로, 문제는 POMDP 로 정의된다. QMIX 는 이러한 POMDP 환경에서 중앙집중적 학습과 분산적 실행(centralized training with decentralized execution, CTDE) 구조를 사용한다.

학습 시에는 중앙 mixing network 가 모든 에이전트의 상태와 개별 Q 값을 집계해 전역 Q 값을 산출하고, 실행 시에는 각 에이전트가 자신의 로컬 관찰만을 이용해 요청을 지연시킬지 혹은 이웃한 노드로 라우팅할지 등의 행동을 선택한다. 보상 함수는 요청을 최종 목적지까지 성공적으로 전달하면 +1, 실패 또는 lifetime 초과 시 0 을 부여한다. 환경의 목적은 제한 시간 내 요청을 최대한 많이 처리하는 것이므로 큐잉 및 처리 지연은 보상 체계에서 제외한다.

3. 비교 알고리즘과 hop-by-hop 의 차별성

성능 비교군으로 구현한 SP 는 요청 발생 시점에 전역 최단 경로를 계산하여 경로 상의 링크를 통해 키를 할당한다. WSP 는 DARPA QKD 네트워크에서 제안된 방식으로 [4], 각 링크의 남은 키 블록 수에 기반해 링크 metric 을 계산하고, 남은 키가 적은 링크에는 큰 비용을 부여해 경로 계산 시 회피하도록 설계하였다. 이 두 방식은 모두 요청 시점에 한 번만 경로를 계산하므로, 경로 중간의 링크에서 키가 고갈되면 전체 요청을 폐기해야 한다. 반면 제안한 MARL 기반 hop-by-hop 라우팅은 각 홉에서 다음 홉을 실시간으로 재선택하여, 중간 링크의 키 고갈로 인한 요청 실패를 방지하고, 요청 부하를 자연스럽게 분산시킬 수 있다.

또한 본 연구에서는 SP 와 WSP 에 hop-by-hop 방식을 적용한 HSP 와 HWSP 를 함께 구현하였다. 이들은 각 홉에서 인접 링크들의 상태를 실시간으로 평가하여 다음 홉을 선택함으로써, 경로 전체를 사전에 고정하지 않고 동적으로 경로를 구성한다. 이러한 동작은 기존의 전역 경로 기반 소스 라우팅 기법들이 트래픽 변화에 취약하다는 한계를 극복하며, 자원 희소성이 큰 QKDN 환경에서 특히 유리하다.

Ⅲ. 결 론

본 논문은 QKD 네트워크에서 요청의 스케줄링과 라우팅 문제를 해결하기 위해 hop-by-hop 방식의 다중 에이전트 강화학습 기반 제어 프레임워크를 제안한다. (그림 2)를 통해 알 수 있듯 본 연구는 hop-by-hop 분산 제어와 MARL 접근 방식이 QKD 라우팅의 유망한 대안임을 실험적으로 검증하였다. Grid **3×3** 에서 QMIX는 HWSP 대비 실패율이 4% 감소하였다. 향후에는 실제 및 대규모 토폴로지에 적용하여 확장성과 일반화 성능을 평가할 계획이다.

ACKNOWLEDGMENT

이 논문은 2025 년도 한국과학기술정보연구원(KISTI)의 기본사업의 지원(과제번호: (KISTI)K25L5M2C2)과 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원(No.RS-2025-02263666)을 받아 수행된 연구임.

참 고 문 헌

[1] Ioannidis, S., et al., “Jointly Optimal Routing and Caching for Arbitrary Network Topologies,” IEEE Journal on Selected Areas in Communications, pp. 1-1, 2018.

[2] Rashid, T., et al., “QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning,” ICML, 4295-4304, 2018.

[3] Coopmans, T., Knegjens, R., Dahlberg, A. et al., “NetSquid, a NETwork Simulator for QUantum Information using Discrete events,” Communications Physics, vol. 4, pp. 164, 2021.

[4] Mehic, M., et al., “Quantum key distribution: A networking perspective,” ACM Computing Surveys, vol. 53, no. 5, 2020.