

MQTT 기반 IoT 네트워크에서 강화학습을 활용한 Retained 메시지 전송 방법

경연웅¹, 김태국², 김영준^{3*}

¹국립공주대학교 정보통신공학과 교수, ²국립부경대학교 컴퓨터공학부 교수, ³경남대학교 컴퓨터공학부 교수

Retained Message Delivery Scheme utilizing Reinforcement Learning in MQTT-based IoT Networks

Yeunwoong Kyung¹, Tae-Kook Kim², Youngjun Kim^{3*}

¹Professor, Division of Information & Communication Engineering, Kongju National University

²Professor, School of Computer and Artificial Intelligence Engineering, Pukyong National University

³Professor, School of Computer Science and Engineering, Kyungnam University

요약 MQTT 프로토콜에서 Publisher로부터 발행되는 메시지의 retained flag가 세팅되어 있으면 해당 메시지는 Broker에 Retained 메시지로 저장되고, 새로운 Subscriber가 subscribe를 수행할 때 Broker는 Retained 메시지를 바로 전송하게 된다. 이를 통해 새로운 Subscriber는 Publisher의 새로운 메시지 발행을 기다리지 않고 Retained 메시지를 통해 현재 상태에 대한 업데이트를 수행할 수 있다. 하지만 Publisher로부터 새로운 메시지가 자주 발행되는 경우에는 retained 메시지를 보내는 것이 트래픽의 오버헤드가 될 수 있고, 해당 상황은 새로운 Subscriber들의 subscribe가 자주 수행되는 경우 더욱 큰 오버헤드로 고려될 수 있다. 그러므로 본 연구에서는 이러한 문제를 해결하기 위해 발행되는 메시지의 특성을 고려하여 Broker의 Retained 메시지 전송 방법을 제안하고자 한다. 본 연구에서는 Broker 입장에서 새로운 Subscriber로의 전송 및 대기 액션을 고려하여 강화학습을 기반으로 모델링하였고, Q learning 알고리즘을 통해 최적의 전송 방법을 결정하였다. 성능 분석을 통해 제안하는 방법이 기존 방법 대비 개선된 성능을 보이는 것을 확인하였다.

주제어 : MQTT, Retained 메시지, 강화학습, Q learning, 최적 정책

Abstract In the MQTT protocol, if the retained flag of a message published by a publisher is set, the message is stored in the broker as a retained message. When a new subscriber performs a subscribe, the broker immediately sends the retained message. This allows the new subscriber to perform updates on the current state via the retained message without waiting for new messages from the publisher. However, sending retained messages can become a traffic overhead if new messages are frequently published by the publisher. This situation could be considered an overhead when new subscribers frequently subscribe. Therefore, in this paper, we propose a retained message delivery scheme by considering the characteristics of the published messages. We model the delivery and waiting actions to new subscribers from the perspective of the broker using reinforcement learning, and determine the optimal policy through Q learning algorithm. Through performance analysis, we confirm that the proposed method shows improved performance compared to existing methods.

Key Words : MQTT; Retained Message; Reinforcement Learning; Q Learning, Optimal Policy

This research was supported by "Regional innovation Strategy (RIS)" through the National Research Foundation of Korea (NRF) funded by the Ministry of Education(MOE)(2021RIS-003)

*교신저자 : 김영준(youngjun@kyungnam.ac.kr)

접수일 2024년 02월 24일 수정일 2024년 03월 11일 심사완료일 2024년 03월 22일

1. 서론

IoT (Internet of Things)의 발전으로 IoT 지원을 위한 다양한 프로토콜들이 소개되고 도입되고 있다[1,2]. 이 중 가볍고 에너지 효율적인 장점으로 인해 MQTT (Message Queuing Telemetry Transport) 프로토콜이 de-facto 표준으로 고려되고 있다[3]. MQTT는 스마트 제조 및 홈 오토메이션 등 다양한 분야에서 사용되고 있으며 AWS (Amazon Web Service) 및 GCP (Google Cloud Platform) 등 다양한 클라우드 플랫폼에서도 지원되고 있다[4-6].

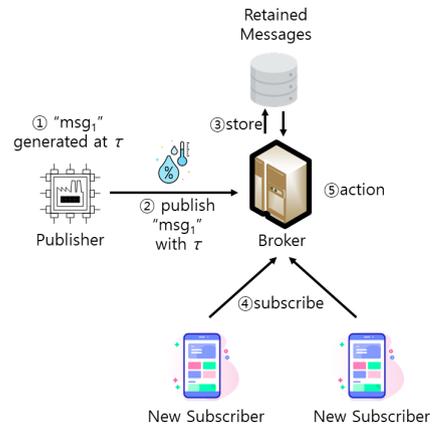
MQTT는 Fig. 1과 같이 메시지를 발행하는 Publisher와 Publisher를 subscribe하는 Subscriber, 그리고 Publisher의 메시지를 Subscriber에게 전달해주는 Broker로 구성된다[7]. 즉, Publisher의 모든 메시지는 Broker를 통해 Subscriber에게 전달되는 구조이다. Publisher에서 새로운 메시지가 발생되어 Broker로 전달할 때, Publisher는 해당 메시지에 retained flag를 세팅해서 전송할 수 있고, 해당 메시지는 Broker에 Retained 메시지로 저장된다. 이후 새로운 Subscriber가 subscribe를 수행하게 되면 Broker는 저장되어 있는 Retained 메시지를 바로 Subscriber에게 전송하게 된다. 현재 MQTT 프로토콜은 Fig. 1의 5번 action 부분은 Retained 메시지를 바로 전송하는 것에 대해서만 정의하고 있다. 즉, Subscriber는 Publisher로부터 새로운 메시지의 발행을 기다리지 않고 subscribe를 하자마자 Retained 메시지를 수신하여 해당 메시지를 기반으로 상태를 업데이트할 수 있다. 하지만 이러한 Retained 메시지 동작은 Publisher로부터 새로운 메시지가 자주 발생하는 경우에는 트래픽의 오버헤드가 될 수 있다. 또한 해당 상황은 새로운 Subscriber들의 subscribe가 자주 수행되는 경우 더욱 큰 오버헤드로 고려될 수 있다. Retained 메시지 관련 기존 연구로는 Broker에 두 개 이상의 Retained 메시지를 저장하고자 하는 저장소 관련 연구[8]가 있었지만 트래픽의 오버헤드를 고려하지는 못하였다. 그러므로 본 연구에서는 이러한 문제를 해결하기 위해 발생하는 메시지의 특성을 고려하여 Broker의 Retained 메시지 전송 방법을 제안하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 제안하는 Retained 메시지 전송 방법의 모델링을 제시하고, 3장에서는 성능 분석을 진행한다. 그리고 4장에서 결론을 제시하며 마무리하고자 한다.

2. Markov Decision Process (MDP)

모델링

Fig. 1에서 제시한 시스템 모델에서 강화학습을 기반으로 Retained 메시지 전송의 최적 정책을 결정하기 위해 Markov Decision Process (MDP) 모델을 구성하면 다음과 같이 상태, 액션, 전이확률, 보상 및 비용 함수를 정의할 수 있다[9-12].



[Fig. 1] MQTT Operation with New Subscribers

2.1 상태

본 모델에서의 상태 집합 \mathcal{S} 는 다음과 같이 정의될 수 있다.

$$\mathcal{S} = \mathcal{U} \times \mathcal{T} \times \mathcal{D} \times \mathcal{N} \quad (1)$$

위의 식에서 \mathcal{U} 는 Publisher로부터 Broker에 전달되는 새로운 메시지의 유무를 의미하고, \mathcal{D} 는 Broker가 메시지 전송을 지연시키는 시간, \mathcal{T} 는 메시지가 Subscriber에게 전달되는 시간, 그리고 \mathcal{N} 은 Subscriber의 수를 의미한다.

먼저 \mathcal{U} 는 새로운 메시지의 유무이기 때문에 $\{0,1\}$ (즉, 0과 1은 각각 새로운 메시지가 없는 상태와 있는 상태를 의미한다.)으로 정의할 수 있다. \mathcal{D} 의 경우 0부터 최대 지연 시간인 D_{max} 까지의 이산 값을 포함하는 집합으로 정의하였다. 또한 \mathcal{T} 와 \mathcal{N} 의 경우 모두 0 이상의 이산 값을 포함하는 집합으로 가정하였다.

2.2 액션

본 모델에서의 액션 집합 \mathcal{A} 는 $\{0,1\}$ 로 정의하였다. 즉, 0의 의미는 Broker가 현재 MQTT 프로토콜에서 정의된 바와 같이 새로운 Subscriber가 subscribe 수행 시 Retained 메시지를 바로 해당 Subscriber에게 전송하

는 것이고, 1의 의미는 Broker가 메시지 전송을 지연시키는 것을 의미한다.

2.3 전이확률

본 모델에서의 전이확률 $P[s'|s,a]$ 는 현재 상태 $s(\in \mathcal{S})$ 에서 액션 $a(\in \mathcal{A})$ 를 선택했을 때, 다음 상태 $s'(\in \mathcal{S})$ 으로 가는 확률을 의미한다. 2.1에서 정의한 상태 집합 \mathcal{S} 에서 \mathcal{U} 와 \mathcal{N} 은 액션 및 다른 상태들과 독립적으로 전이되고, \mathcal{D} 와 \mathcal{T} 의 경우 액션에 의존적이기 때문에 $P[s'|s,a]$ 는 $P[u'|u] \times P[n'|n] \times P[d'|d,a] \times P[t'|t,a]$ 로 정의될 수 있다. 본 논문에서는 Publisher로부터 메시지가 발행되어 Broker로 전송되는 확률이 평균 λ_p 를 갖는 Poisson 분포를 따른다고 가정하였고, Subscriber가 Publisher를 구독(해지)할 확률은 평균 $\lambda_s(\lambda_u)$ 갖는 Poisson 분포를 따른다고 가정하였다[5]. 또한 $P[d'|d,a]$ 의 경우 Broker가 액션 1을 선택했을 때 최대 지연 시간인 D_{max} 까지 1씩 증가하고 (즉, $d' = d+1$), action 0을 선택했을 때는 $d' = d$ 가 된다. 마지막으로 $P[t'|t,a]$ 의 경우 Broker가 액션 1을 선택했다면 $t' = t+1$ 이 되고, action 0을 선택했을 경우 새로운 Subscriber에게 전송되기 때문에 Publisher가 발행할 때 메시지 내에 입력한 해당 메시지의 발생 시간부터 Subscriber의 수신시간까지의 지연 시간을 t' 으로 계산한다[13-15].

2.4 보상 및 비용함수

본 모델에서의 보상 및 비용함수 $r(s,a)$ 는 메시지 전달을 위한 지연 시간 및 트래픽 오버헤드를 고려하여 다음과 같이 정의한다.

$$r(s,a) = \omega g(s,a) - (1-\omega)f(s,a) \quad (2)$$

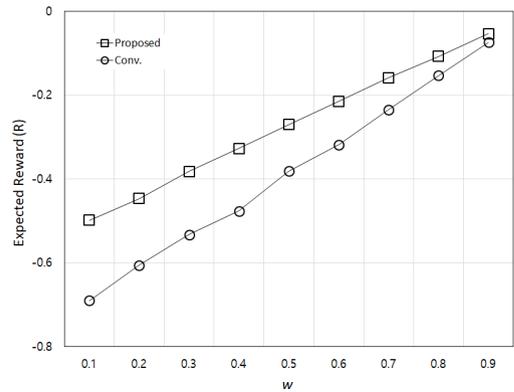
위의 수식에서 $g(s,a)$ 는 지연 시간을 고려한 보상함수를 의미하고 $f(s,a)$ 는 트래픽 오버헤드를 고려한 비용함수를 의미한다. 그리고 ω 는 $g(s,a)$ 와 $f(s,a)$ 의 가중치를 의미한다. 구체적으로 $g(s,a)$ 는 2.3에서 설명한 바와 같이 Broker에서 각 액션을 선택했을 때 계산되는 t' 으로 결정된다. $f(s,a)$ 의 경우 새로운 메시지가 Broker에 수신된 경우 ($u=1$) 모든 Subscriber에게 해당 메시지를 전달해야 하기 때문에 $n \cdot C_p$ (n 은 2.1에서 정의한 Subscriber의 수, C_p 는 하나의 메시지 전달 비용)로 계산될 수 있다. 또한 새로운 메시지가 수신되지 않은 상황에서 새로운 Subscriber가 발생했을 때, $f(s,a)$ 는 액션 0을 선택한 경우 해당 Subscriber에 메시지를 전달해야 하기 때문에 C_p 로 계산될 수 있고, 액션 1을 선택할 경우 지연을 시키는 것이기 때문에 0으로 계산된다.

2.5 최적화

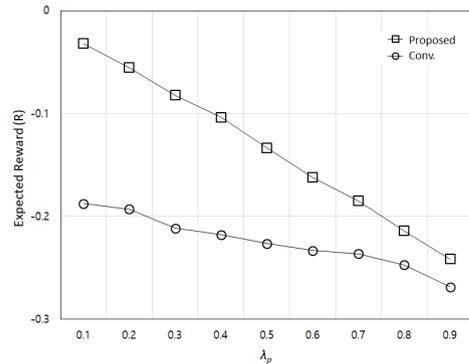
본 논문에서는 Q 러닝 알고리즘을 통해 최적의 정책을 찾고자 한다. 즉, 아래와 같이 상태-액션 가치함수인 $Q(s,a)$ 업데이트를 통해 Q 테이블을 구성하고, 각 상태에서 $Q(s,a)$ 를 최대화 시키는 액션을 선택할 수 있다[5].

$$Q(s,a) \leftarrow Q(s,a) + \alpha(R + \theta \max_{a'} Q(s',a') - Q(s,a)) \quad (3)$$

위 식에서 α 는 learning rate, R 은 즉각적인 보상, 그리고 θ 는 discount factor를 의미한다.



[Fig. 2] Expected reward according to ω



[Fig. 3] Expected reward according to λ_p

3. 성능분석

성능 분석을 위해 본 논문에서는 Python 기반의 시뮬레이터를 구현하여 시뮬레이션을 수행하였고, 제안하는 방법(Proposed)과 기존 MQTT 표준 방식(Conv.)에 대한 성능 분석을 진행하였다. 성능분석을 위한 디폴트 파라미터 값은 다음과 같다. λ_p 는 0.6이고, $\lambda_s(\lambda_u)$ 은

0.2(0.1), ω 는 0.7, C_p 는 5로 설정하였다. 또한 알고리즘 관련하여 α 는 0.2, θ 는 0.95로 설정하였다.

Fig. 2는 ω 가 변할 때 제안하는 방법(Proposed)과 기존 방법(Conv.)의 보상 및 비용의 평균값(Expected Reward)을 보여준다. Fig. 2에서 알 수 있듯이 기존 방법은 Retained 메시지를 전송하는 액션만 정의되어 있기 때문에 ω 가 작아짐에 따라 트래픽 오버헤드로 인해 보상 및 비용의 평균값이 제안 방법과 대비하여 상대적으로 더 많이 감소하는 것을 확인할 수 있다. 반면 제안하는 방법은 ω 가 작아져도 전송을 지연하는 액션이 있기 때문에 보상 및 비용의 평균값이 기존 방법 대비 상대적으로 덜 감소하는 것을 확인할 수 있다.

Fig. 3는 λ_p 가 변할 때 제안하는 방법(Proposed)과 기존 방법(Conv.)의 보상 및 비용의 평균값(Expected Reward)을 보여준다. Fig. 3에서 알 수 있듯이 λ_p 가 작을 때는 제안하는 방법의 경우 전송을 지연시키는 액션으로 인해 Retained message로 인한 트래픽 오버헤드를 줄일 수 있어서 기존 방법 대비 상대적으로 높은 보상 및 비용의 평균 값을 갖는 것을 확인할 수 있다. λ_p 가 클 때는 새로운 메시지의 잦은 발생으로 인해 모든 subscriber에게 publish 메시지를 자주 발생해야 해서 제안하는 방법과 기존 방법 모두 낮은 보상 및 비용의 평균값을 갖는 것을 확인할 수 있다.

4. 결론 및 향후 연구 방안

본 논문은 MQTT 프로토콜에서 publisher로부터 발행되는 메시지의 특성을 고려하여 Broker의 Retained 메시지 전송 방법을 제안하였다. MDP를 기반으로 모델링을 수행하였고 Q 러닝 알고리즘을 기반으로 최적의 정책을 결정하였다. 성능 분석을 통해 보상 및 비용의 평균값 측면에서 기존 MQTT 프로토콜 대비 제안하는 방법의 우수성을 입증하였다. 향후에는 MQTT 기반 네트워크의 규모가 확장되었을 때, 심층 강화학습을 기반으로 제안하는 방법을 모델링하고 효율성을 검증하고자 한다.

REFERENCES

[1] D.W.Lee, K.Cho and S.H.Lee, "Analysis on Smart Factory in IoT Environment," *Journal of The Korea Internet of Things Society*, Vol.5, No.2, pp.1-5, 2019.

[2] C.Choi., "A Study On IoT Data Consistency in IoT Environment," *Journal of The Korea Internet of*

Things Society, Vol.8, No.5, pp.127-132, 2022.

- [3] F.Buccafurri et al., "MQTT-I: Achieving end-to-end data flow integrity in MQTT," *IEEE Trans. on Dependable and Secure Computing*, pp.1-18, 2024.
- [4] H.Shi et al., "Construction of industrial internet of things based on MQTT and OPC UA protocols," in *Proc. ICAICA*, pp.1263-1267, 2020.
- [5] Y.Kyung et al., "Priority-aware actuation update scheme in heterogeneous industrial networks," *Sensors*, Vol.24, 2024.
- [6] K.Govindan et al., "End-to-end service assurance in IoT MQTT-SN," in *Proc. IEEE CCNC*, 2015.
- [7] "Information technology-message queuing telemetry transport (mqtt)," International Organization for Standardization, iso.org, Standard, 2016.
- [8] X.Wu et al., "Improvements of MQTT retain message storage mechanism," in *Proc. IMCEC*, 2018.
- [9] Y.Kyung et al., "Opportunistic offloading scheme for content delivery service using electro-mobility networks," *IET Intelligent Transport System*, 2022.
- [10] X.Wang, C.Chen, J.He, S.Zhu and X.Guan, "Aoi-aware control and communication co-design for industrial iot systems," *IEEE Internet of Things Journal*, vol. 8, no. 10, pp. 8464-8473, 2021.
- [11] Y.Sun, E.Uysal-Biyikoglu, R.D.Yates, C.E.Koksal and N.B.Shroff, "Update or wait: How to keep your data fresh," *IEEE Transactions on Information Theory*, vol. 63, no. 11, pp. 7492-7508, 2017.
- [12] F.Palmese, A.E.C.Redondi and M.Cesana, "Adaptive quality of service control for mqtt-sn," *Sensors*, vol. 22, no. 22, 2022.
- [13] L.Hu, Z.Chen, Y.Dong, Y.Jia, L.Liang and M.Wang, "Status update in iot networks: Age-of-information violation probability and optimal update rate," *IEEE Internet of Things Journal*, vol. 8, no. 14, pp. 11329-11344, 2021.
- [14] Y.Chiang, H.Lin and Y.Ji, "Information cofreshness-aware grant assignment and transmission scheduling for internet of things," *IEEE Internet of Things Journal*, vol. 8, no. 19, pp. 14435-14446, 2021.
- [15] R.O.Aburukba, M.AliKarrar, T.Landolsi, and K.El-Fakih, "Scheduling internet of things requests to minimize latency in hybrid fog-cloud computing," *Future Generation Computer Systems*, vol. 111, pp. 539-551, 2020

경 연 웅(Yeunwoong Kyung) [종신회원]



- 2011년 2월 : 고려대학교 전기전자전파공학부(공학사)
- 2016년 8월 : 고려대학교 전기전자전파공학부(공학박사)
- 2016년 9월 ~ 2020년 3월 : 삼성전자 무선사업부 책임연구원

- 2020년 3월 ~ 2022년 8월 : 한신대학교 컴퓨터공학부 교수
- 2022년 9월 ~ 현재 : 국립공주대학교 정보통신공학과 교수

〈관심분야〉

사물인터넷(IoT), SDN, 5G/6G, 이동성, 모바일 서비스

김 태 국(Tae-Kook Kim) [종신회원]



- 2004년 8월 : 고려대학교 전기전자전파공학부(공학사)
- 2006년 8월 : 고려대학교 메카트로닉스학과(공학석사)
- 2014년 8월 : 고려대학교 모바일솔루션학과(공학박사)
- 2016년 3월 ~ 2022년 2월 : 동명대학교 AI학부 교수

- 2022년 3월 ~ 현재 : 국립부경대학교 컴퓨터·인공지능공학부 교수

〈관심분야〉

사물인터넷(IoT), 콘텐츠 전송 네트워크(CDN), 이동성, 인공지능(AI), 빅데이터(Big Data), 모바일 서비스

김 영 준(Youngjun Kim) [정회원]



- 2010년 2월 : 고려대학교 전기전자전파공학부(공학사)
- 2022년 8월 : 고려대학교 전기전자전파공학부(공학박사)
- 2022년 9월 ~ 현재 : 경남대학교 컴퓨터공학부 교수

〈관심분야〉

사물인터넷(IoT), WiFi, SDN, 모바일 서비스