Delay-reliability-aware protocol adaption and quality of service guarantee for message queuing telemetry transport-empowered electric Internet of things

Hao Zhang1, Hongshuo Zhang1, Zhao Wang1, Zhenyu Zhou1, Qiang Wang2, Guangyuan Xu3, Junzhong Yang4 and Zhong Gan5

Abstract
Message queuing telemetry transport has emerged as a promising communication protocol for resource-constrained electric Internet of things due to high bandwidth utilization, simple implementation, and various quality of service levels. Enabled by message queuing telemetry transport, electric Internet of things gateways adopt dynamic protocol adaptation, conversion, and quality of service level selection to realize bidirectional communication with massive devices and platforms based on heterogeneous communication protocols. However, protocol adaptation and quality of service guarantee in message queuing telemetry transport-empowered electric Internet of things still faces several challenges, such as unified communication architecture, differentiated quality of service requirements, lack of quality of service metric models, and incomplete information. In this paper, we first establish a unified communication architecture for message queuing telemetry transport-empowered electric Internet of things for adaptation and conversion of heterogeneous protocols. Second, we formulate the quality of service level selection optimization problem to minimize the weighted sum of packet-loss ratio and delay. Then, a delay-reliability-aware message queuing telemetry transport quality of service level selection algorithm based on upper confidence bound is proposed to learn the optimal quality of service level through dynamically interacting with the environment. Compared with single and fixed quality of service level selection strategies, delay-reliability-aware message queuing telemetry transport quality of service level selection can effectively reduce the weighted sum of delay and packet-loss ratio and satisfy the differentiated quality of service requirements of electric Internet of things.

Keywords
Message queuing telemetry transport, electric Internet of things, protocol adaption, quality of service guarantee, delay and reliability awareness, upper confidence bound algorithm, dynamic quality of service level selection

Introduction
Electric Internet of things (EIoT) can provide significant support for the intelligence, digitalization, and transparency of power grid through timely collecting the operation parameters, including voltage, current, as well as active and reactive power, and transmitting

Date received: 22 December 2021; accepted: 17 March 2022
Handling Editor: Peio Lopez Iturri

Corresponding author:
Zhao Wang, State Key Laboratory of Alternate Electrical Power System with Renewable Energy Sources, North China Electric Power University, Beijing 102206, China.
Email: zhao_w@ncepu.edu.cn
them to the cloud platform for processing and analysis. In EIoT, the communication devices produced by different manufacturers utilize multiple communication protocols for data transmission and information interaction. Typical communication protocols in EIoT include message queue telemetry transport (MQTT), data distribution service (DDS), constrained application protocol (CoAP), hypertext transfer protocol (HTTP), etc. DDS is commonly used for state monitoring in EIoT. CoAP is particularly suitable for services like meter reading management and load forecasting. HTTP is applicable for high-performance devices with large computing and storage resources in EIoT. MQTT is suitable for lightweight data transmission of gateways due to the characteristics of high bandwidth utilization and simple implementation. The gateway can achieve the adaptation and conversion of different protocols to MQTT. Through the information interaction between gateways based on MQTT, the connectivity and interoperability among different devices can be achieved, which shields the differences among various protocols.

QoS guarantee is of vital importance in the process of data transmission between gateway and platform in EIoT. MQTT provides three quality of service (QoS) levels, that is, at most once (QoS0) level, at least once (QoS1) level, and exactly once (QoS2) level, which provide different QoS guarantees in terms of transmission delay and packet-loss ratio. Specifically, the transmission delay of QoS0 is relatively lower but the packet-loss ratio is higher, while QoS1 and QoS2 achieve no packet loss at the expense of increased transmission delay. Moreover, QoS1 guarantees that the data packet is successfully transmitted at least once, and QoS2 ensures that the data packet is successfully transmitted exactly once by leveraging more complicated retransmission mechanism. Therefore, it is necessary to dynamically and intelligently select MQTT QoS levels for data transmission between gateway and platform according to the time-varying network state and QoS requirements in EIoT.

However, the dynamic MQTT QoS level selection still faces some challenges, which are summarized as follows. First, the QoS requirements of control services and acquisition services differ in terms of delay and reliability. However, the different metrics are contradictory, for example, adopting retransmission mechanism ensures lower packet-loss ratio but greatly increasing transmission delay. Therefore, it is a critical challenge to achieve a balanced trade-off among different QoS metrics. Second, the current delay and packet-loss ratio models do not take the impact of protocol-specific QoS guarantee mechanism on the physical-layer performance into consideration. Therefore, deriving the accurate closed-form models of delay and packet-loss ratio which are adaptive with MQTT-specific QoS levels is challenging. Last but not least, due to network resource limitation and prohibitive signaling overhead, the global state information (GSI), for example, channel gain, is uncertain. Therefore, it is necessary to intelligently optimize MQTT QoS level selection under incomplete information.

There exist some works that have addressed MQTT QoS level selection problems in IoT. Sadeq et al. proposed a QoS approach for IoT environment utilizing MQTT and designed a flow control mechanism to minimize the transmission delay. Niruntasukrat et al. proposed an authorization mechanism for MQTT-based IoT service platform to minimize delay and message overhead. However, these works have not considered the joint optimization of delay and packet-loss ratio. Lee et al. proposed a push notification service network utilizing MQTT protocol to minimize the packet loss and delay by selecting appropriate QoS level according to different payloads. Nurwarsito et al. proposed a communication architecture using MQTT protocol for emergency vehicles which aims to minimize the packet loss and average delay. However, the above-mentioned works have not considered uncertain GSI in practical IoT application scenarios. Weerasinghe et al. proposed an MQTT-based localization mechanism for wireless sensor network by utilizing supervised learning. Ahmadon et al. proposed a machine learning-based anomaly detection method for MQTT-based network. However, these works need offline scene data, which cannot adapt to the complex environment in EIoT.

Reinforcement learning provides a powerful tool to deal with sequential decision problems under incomplete information. Among various reinforcement learning algorithms, upper confidence bound (UCB) originally developed for the multi-armed bandit (MAB) problems has rapid convergence speed and well-balanced trade-off between exploitation and exploration. Zhou et al. proposed an energy-aware and data backlog-aware UCB-based channel selection algorithm, which can improve energy efficiency and throughput. However, the delay and reliability are not taken into account. Endo et al. proposed a distributed QoS-UCB channel selection algorithm considering channel rating quality, which can improve the reliability and reduce the delay while avoiding congestion. However, this work has not considered the complex communication environment in EIoT and MQTT-specific QoS level selection optimization.

Motivated by the aforementioned challenges, we propose a delay-reliability-aware protocol adaption and QoS guarantee method for EIoT based on reinforcement learning. First, considering the adaptation and conversion of heterogeneous protocols, we establish a communication architecture of EIoT based on MQTT.
Second, we propose a delay-reliability-aware MQTT QoS level selection (DR-MQLS) algorithm based on UCB to minimize the weighted sum of packet-loss ratio and delay. Last but not least, simulations are carried out to validate the effectiveness of DR-MQLS. Compared with single and fixed QoS level selection strategies, DR-MQLS can effectively reduce the weighted sum of packet-loss ratio and delay and satisfy the differentiated QoS requirements in EIoT. We summarize the main contributions of this work as follows:

- **Intelligent QoS Guarantee under Incomplete Information**: DR-MQLS enables gateway to interact with environment and learn the optimal QoS level selection based on UCB under incomplete information. DR-MQLS can realize intelligent QoS guarantee with only local information.
- **Delay and Reliability Awareness**: The closed-form models of delay and packet-loss ratio for three MQTT-specific QoS levels are derived. The optimization objective is defined to minimize the weighted sum of packet-loss ratio and delay. DR-MQLS can achieve delay and reliability awareness by selecting the MQTT QoS levels according to the specific QoS requirements of EIoT services.
- **Extensive Performance Evaluation**: Extensive simulations are carried out to demonstrate the effectiveness and reliability of DR-MQLS. Specially, the effects of various parameter settings, such as the signal-to-noise threshold and the weight of delay in the optimization objective, have been illustrated to provide guidance for practical application.

The remaining structure is as follows. In section “System model and problem formulation,” we describe system model and problem formulation in details. The proposed DR-MQLS algorithm is introduced in section “Delay-reliability-aware MQTT QoS level selection in EIoT.” Section “Simulation results” provides simulation results. In section “Conclusion,” we summarize this article.

**System model and problem formulation**

The considered communication architecture of EIoT based on MQTT is shown in Figure 1, which consists of an MQTT broker server, a cloud platform, multiple EIoT devices, and multiple gateways. The gateways with protocol adaption and conversion functions adopt publish/subscribe pattern for information interaction with cloud platform and can act as both publishers and subscribers. The broker server acts as an intermediary for data transmission between publishers and subscribers, which is deployed on the cloud platform. The publisher notifies the broker server with topics which it tends to publish. Then, the broker server keeps the topics and pushes them when subscribers ask for relevant topics. Multiple communication protocols are used for data transmission between EIoT devices and gateways, for example, HTTP, CoAP, and DDS. Through parsing and repackaging protocol messages, the gateway achieves the conversion between...
multiple protocols and MQTT protocol. An example is shown in Figure 1. The broker server pushes the subscribed topic and transmits the related data to the gateway based on the transmission mechanism specified by MQTT QoS1 level. Then the gateway executes protocol adaption and conversion to repackage protocol messages based on DDS, CoAP, and HTTP and transmits the data to the corresponding EIoT devices.

We assume that there are $I$ large packets to be transmitted, and the set is $I = \{1, \ldots, i, \ldots, I\}$. Each large packet consists of $J$ small packets, and the set is $J = \{1, \ldots, j, \ldots, J\}$. The selected QoS level remains unchanged within a large packet but varies across different large packets. Denote the three QoS levels as $m = 0, 1, 2$, respectively. Define $x_{m}^{i} \in \{0, 1\}$ as the MQTT QoS level selection variable, where $x_{m}^{i} = 1$ represents that the $m$th QoS level is selected by the $i$th large packet, and $x_{m}^{i} = 0$ otherwise.

We assume that channel state remains unchanged during small packet data transmission process but varies across different small packets. In particular, each retransmission is considered as a small packet transmission process for QoS1 and QoS2 which adopt retransmission mechanisms. The channel gain $G_{i,j,n}$ of the $n$th transmission of the $j$th small packet of the $i$th large packet is given by

$$G_{i,j,n} = \frac{|H_{i,j,n}|^2}{N_0 + e_{i,j,n}}$$  \hspace{1cm} (1)

where $H_{i,j,n}$ represents the channel frequency response. $N_0$ represents the noise power. $e_{i,j,n}$ is the electromagnetic interference power. Since each message in QoS0 level is only transmitted once, we define $n = 1$ in QoS0 level.

Figure 2 shows MQTT data transmission processes of three QoS levels. The packet-loss ratio and delay models of the three QoS levels are elaborated in the following.

**QoS0 level**

QoS0 provides best-effort delivery of the PUBLISH packet. After the gateway sending the PUBLISH packet to the broker server, the transmission process is completed immediately, regardless of whether the broker server receives the packet. Therefore, although the transmission delay of QoS0 is low, the packet-loss ratio is relatively high under poor channel states.

**Packet-loss ratio model.** QoS0 level for data transmission has only one PUBLISH packet transmission process. Therefore, the packet-loss variable of the $j$th small packet of the $i$th large packet in QoS0 level is given by

$$d_{0,i,j} = \begin{cases} 1, & \text{if } PG_{i,j,1} < G_{th} \\ 0, & \text{if } PG_{i,j,1} \geq G_{th} \end{cases}$$  \hspace{1cm} (2)

where $G_{th}$ represents the signal-to-noise ratio threshold. $P$ represents the transmission power. If the current signal-to-noise ratio $PG_{i,j,1}$ is lower than the threshold $G_{th}$, the PUBLISH packet of the $j$th small packet of the $i$th large packet is lost, that is, $d_{0,i,j} = 1$. Otherwise, $d_{0,i,j} = 0$. Therefore, in QoS0 level, the packet-loss ratio of the $i$th large packet is given by

$$Q^i = \frac{d_{0,i}}{J}$$  \hspace{1cm} (3)

Here, $d_{0,i}$ represents the number of lost packet of the $i$th large packet, which is given by

$$d_{0,i} = \sum_{j=1}^{J} d_{0,i,j}.$$
Delay model. The transmission delay of the $j$th small packet of the $i$th large packet in QoS0 level is given by

$$t_{i,j}^L = \frac{U(i)}{B \log_2 (1 + PG_{i,j,1})}$$

where $U(i)$ represents the packet size of each small packet of the $i$th large packet. $B$ represents the bandwidth.

The total delay of the $i$th large packet is given by

$$T_i^0 = \sum_{j=1}^{J} t_{i,j}^L$$

QoS1 level

QoS1 adopts a PUBACK packet to acknowledge the reception of the PUBLISH packet. If the PUBACK packet is not received by the gateway within a certain time, the PUBLISH packet is retransmitted. In this case, the PUBLISH packet is received at least once at the broker server. The data deduplication process is required to delete the duplicate packets at the expense of a certain data processing delay. Therefore, the packet-loss ratio in QoS1 level is zero, but the transmission delay and data deduplication delay are relatively high.

Packet-loss ratio model. Since QoS1 adopts retransmission to ensure successful data transmission, the packet-loss ratio of the $i$th large packet is $Q_i^1 = 0, \forall i \in I$.

Delay model. There are two transmission processes in QoS1 level, that is, PUBLISH packet transmission and PUBACK packet feedback. When the above two processes are successful, the transmission process of a small packet is completed. Define $a_{i,j,n}^r \in \{0, 1\}$ as the $n$th transmission result variable of the PUBLISH packet of the $i$th small packet of the $j$th large packet in QoS1 level. If the current signal-to-noise ratio $PG_{i,j,n}$ is lower than $G_{th}$, the $n$th transmission of the PUBLISH packet is lost, which is denoted as $a_{i,j,n}^r = 1$. Otherwise, $a_{i,j,n}^r = 0$. Therefore, $a_{i,j,n}^r$ is given by

$$a_{i,j,n}^r = \begin{cases} 1, & \text{if } PG_{i,j,n} < G_{th} \\ 0, & \text{if } PG_{i,j,n} \geq G_{th} \end{cases}$$

We define $b_{i,j,n}^1 \in \{0, 1\}$ as the $n$th transmission result variable of the PUBACK packet of the $j$th small packet of the $i$th large packet in QoS1 level. $P_{back}$ is defined as the transmission power of the feedback PUBACK packet. $G_{i,j,n,back}$ is defined as the channel gain of packet feedback. If the feedback signal-to-noise ratio $P_{back}G_{i,j,n,back}$ is lower than $G_{th}$, the $n$th transmission of the PUBACK packet is lost, which is denoted as $b_{i,j,n}^1 = 1$. Otherwise, $b_{i,j,n}^1 = 0$. Therefore, $b_{i,j,n}^1$ is given by

$$b_{i,j,n}^1 = \begin{cases} 1, & \text{if } PG_{i,j,n,back} < G_{th} \\ 0, & \text{if } PG_{i,j,n,back} \geq G_{th} \end{cases}$$

Then, the transmission delay of the $j$th small packet of the $i$th large packet in QoS1 level is given by

$$t_{i,j}^L = \frac{N_{i,j}}{U(i) + t_0} + \frac{1}{B \log_2 (1 + PG_{i,j,n})} + \frac{1}{U(i) + t_0} + \frac{1}{B \log_2 (1 + P_{back}G_{i,j,n,back})}$$

where $N_{i,j}$ represents the total number of transmissions for delivering the $j$th small packet of the $i$th large packet. $U_{ACK}$ represents the packet size of each PUBACK packet. $t_0$ represents the maximum waiting time for the feedback packets. The first term indicates the transmission delay when the PUBLISH packet transmission is failed. The second term indicates the transmission delay when the PUBACK packet transmission is failed. The third and fourth terms indicate the transmission delay when the PUBLISH packet and the PUBACK packet are successfully transmitted, respectively.

In order to simplify the model, we assume that the data deduplication delay of different small packets is uniformly defined as $t_r$. Therefore, the data deduplication delay of the $j$th small packet of the $i$th large packet is given by

$$t_{i,j}^{d} = \sum_{n=1}^{N_{i,j}} b_{i,j,n}^1t_r$$

The total delay of the $i$th large packet in QoS1 level is given by

$$T_i^1 = \sum_{j=1}^{J} t_{i,j}^L + \sum_{j=1}^{J} t_{i,j}^{d}$$

QoS2 level

QoS2 ensures that messages are delivered exactly once through two interaction processes by means of PUBLISH, PUBREC, PUBREL, and PUBCOMP packets. In the first interaction process, after the
gateway sending the PUBLISH packet to the broker server, if a PUBREC is not received within a certain time, the PUBLISH packet will be retransmitted until the PUBREC packet is successfully received. If a duplicate PUBLISH packet is received at the broker server, it will be deleted immediately. In the second interaction process, when receiving the PUBREC packet, the gateway responds to the broker server with a PUBREL packet and waits for the feedback PUBCOMP packet. Similarly, if the PUBCOMP packet is not received within a certain time, the PUBREL packet will be retransmitted until the PUBCOMP packet is successfully received. Therefore, the QoS2 level ensures that each packet is successfully received without duplication.

Packet-loss ratio model. Since QoS2 also adopts retransmission to ensure successful data transmission, the packet-loss ratio of the $i$th large packet is $Q_i^L = 0, \forall i \in I$.

Delay model. There are four processes, that is, PUBLISH packet transmission, PUBREC packet feedback, PUBREL packet transmission, and PUBCOMP packet feedback in QoS2 level. When the above processes are successful, the transmission of a small packet is completed. The PUBREL packet will be transmitted only after the PUBREC packet is successfully fed back.

In the first interaction process, we define $a_{i,j,n}^2 \in \{0, 1\}$ and $b_{i,j,n}^2 \in \{0, 1\}$ as the transmission result variables of the PUBLISH packet and the PUBREC packet, respectively. In the second interaction process, we define $a_{i,j,n}^{2,\text{REL}} \in \{0, 1\}$ and $b_{i,j,n}^{2,\text{COMP}} \in \{0, 1\}$ as the transmission result variables of the PUBREL packet and the PUBCOMP packet, respectively. The judgment of these variables is similar to that of QoS1 and will not be repeated here.

Therefore, the transmission delay of the $j$th small packet of the $i$th large packet in the first interaction process in QoS2 level is given by

$$ t_{i,j,1}^{2, L} = \sum_{n=1}^{N_{i,j,1}} a_{i,j,n}^2 \left( \frac{U(i)}{B \log_2 (1 + PG_{i,j,n})} + t_0 \right) + \sum_{n=1}^{N_{i,j,1}} b_{i,j,n}^2 \left( \frac{U(i)}{B \log_2 (1 + PG_{i,j,n})} + t_0 \right) + \frac{U(i)}{B \log_2 (1 + PG_{i,j,N_{i,j,1}})} + \frac{U_{\text{REC}}}{B \log_2 (1 + P_{\text{back}} G_{i,j,N_{i,j,1}, \text{back}})} $$

(11)

where $U_{\text{REC}}$ represents the packet size of each PUBREC packet. $N_{i,j,1}$ represents the total number of transmissions for delivering the $j$th small packet of the $i$th large packet in the first interaction process. The first term indicates the transmission delay when the PUBLISH packet retransmission is failed. The second term indicates the transmission delay when the PUBREC packet retransmission is failed. The third and fourth terms indicate the transmission delay when the PUBLISH packet and PUBREC packet are successfully transmitted, respectively.

The transmission delay of the $j$th small packet of the $i$th large packet in the second interaction process in QoS2 level is given by

$$ t_{i,j,2}^{2, L} = \sum_{n=1}^{N_{i,j,2}} a_{i,j,n}^{2,\text{REL}} \left( \frac{U_{\text{REL}}}{B \log_2 (1 + PG_{i,j,n})} + t_0 \right) + \sum_{n=1}^{N_{i,j,2}} b_{i,j,n}^{2,\text{COMP}} \left( \frac{U_{\text{REL}}}{B \log_2 (1 + PG_{i,j,n})} + t_0 \right) + \frac{U_{\text{REL}}}{B \log_2 (1 + P_{\text{back}} G_{i,j,N_{i,j,2}, \text{back}})} $$

(12)

where $U_{\text{REL}}$ and $U_{\text{COMP}}$ represent the packet size of each PUBREL packet and PUBCOMP packet, respectively. $N_{i,j,2}$ represents the total number of transmissions for delivering the $j$th small packet of the $i$th large packet in the second interaction process. Since the packet transmission mechanisms of two interaction processes are similar but only the transmitted packets are different, the explanation of formula (12) is omitted here.

Since there is no data deduplication process in QoS2 level, the total delay of the $i$th large packet is the sum of the transmission delays of the two interaction processes, which is given by

$$ T_i^2 = \sum_{j=1}^{f} t_{i,j,1}^{2, L} + \sum_{j=1}^{f} t_{i,j,2}^{2, L} $$

(13)

Problem formulation

To solve the differentiated QoS guarantee problem in EIoT, the optimization objective is defined to minimize the weighted sum of packet-loss ratio and delay under the QoS level selection constraint. The optimization problem is formulated as

$$ \text{P1} : \min_{\{x_{i,m}^n\}} \sum_{i=1}^{I} \sum_{m=0}^{2} \sum_{n=0}^{2} x_{i,m}^n (Q_i^n + V T_i^m) $$

s.t. $C_1 : x_{i,m}^n \in \{0, 1\}, \ \forall i \in I, \ \forall m \in \{0, 1, 2\}$

$$ C_2 : \sum_{m=0}^{2} x_{i,m}^n = 1, \ \forall i \in I $$

(14)
Phase 2: Estimation and QoS selection

1. Select three QoS levels sequentially and obtain the initial rewards.
2. Calculate the preference of the gateway toward the $m$th QoS level as equation (16).
3. Select the optimal QoS level $m^*$ as equation (17).

Phase 3: Learning

1. Observe the packet-loss result and transmission delay of each small packet.
2. Calculate the reward as equation (15).
3. Update $\tilde{\theta}_i^m$ and $k_i^m$ as equations (18) and (19).

The proposed DR-MQLS algorithm

DR-MQLS estimates the reward based on historical observations and considers estimation uncertainty through the confidence bound based on UCB. Therefore, the gateway estimates its preference toward $m$th QoS level as

$$\tilde{\theta}_i^m = \tilde{\theta}_{i-1}^m + \omega \sqrt{\frac{2 \ln i}{k_i^m}}$$

where $V$ represents the weight of delay in the optimization objective, $C_1$ and $C_2$ guarantee that only one QoS level can be selected during the transmission process of each large packet.

Delay-reliability-aware MQTT QoS level selection in EloT

Problem transformation

MAB is an efficient reinforcement learning tool to cope with the sequential decision problems under incomplete information. It describes a sequence of exploration–exploitation decision-making processes. The MAB model is mainly composed of decision makers, arms, and rewards. In each round, the decision maker selects an arm, and the selected arm will generate a reward. The decision maker aims to maximize its reward by exploiting the empirically optimal arm or exploring non-optimal arms.

In this paper, we transform PI into a MAB problem. The decision maker, arm, and reward are modeled as follows:

- **Decision Maker:** Gateways are defined as the decision makers.
- **Arm:** The three QoS levels of MQTT protocol are abstracted as arms, that is, $m \in \{0, 1, 2\}$.
- **Reward:** The reward of selecting the $m$th QoS level is defined as the reciprocal of the weighted sum of packet-loss ratio and delay, which is given by

$$\theta_i^m = \frac{1}{Q_i^m + V T_i^m}$$

$$m^* = \arg \max_{m \in \{0, 1, 2\}} \tilde{\theta}_i^m$$

Therefore, DR-MQLS draws that $x_i^m = 1$. Then, $\tilde{\theta}_i^m$ and $k_i^m$ are updated as

$$\tilde{\theta}_i^m = \frac{\tilde{\theta}_{i-1}^m k_i^{m-1} + \tilde{\theta}_i^{m-1} x_i^m}{k_i^{m-1} + x_i^m}$$

$$k_i^m = k_i^{m-1} + x_i^m$$
The implementation procedure of the proposed algorithm is summarized in Algorithm 1, which is divided into three phases, as follows:

1. **Initialization:** Initialize all the indicator variables as zero, that is, $a_{m,j}^0 = 0$, $a_{i,j,n}^1 = 0$, $b_{i,j,n}^1 = 0$, $a_{i,j,n}^2 = 0$, $b_{i,j,n}^2 = 0$, $a_{i,j,n}^{2,\text{REL}} = 0$, $a_{i,j,n}^{2,\text{COMP}} = 0$, $x_{i,j}^{m} = 0$, $k_{i,j}^{m} = 0$, $\theta_{i}^{m} = 0$. Then, select three QoS levels sequentially for the first three large packet transmissions to obtain the initial rewards.

2. **Estimation and QoS Selection:** The gateway calculates its preference toward the $m^\text{th}$ QoS level as equation (16) and selects the optimal QoS level $m^\ast$ as equation (17).

3. **Learning:** The gateway observes the packet-loss result and transmission delay of each small packet. Then the packet-loss ratio and delay of each large packet are calculated. Finally, calculate the reward as equation (15), update $\theta_{i}^{m}$ and $k_{i}^{m}$ as equations (18) and (19).

**Complexity analysis**

The computational complexity of DR-MQLS is composed of three parts. The computational complexity of the first phase is $O(7J) + O(9I)$, and that of the second phase is $O(6I)$. The computational complexity of the third phase is $O(J + 8)I$. Therefore, the computational complexity of DR-MQLS is $O(7J) + O(9I) + O(6I) + O(J + 8)I$.

**Simulation results**

In this section, we validate the performance of DR-MQLS through simulations. The single and fixed QoS level selection strategies, that is, only selecting a specific QoS level for data transmission, for example, QoS0, QoS1, and QoS2, are used for comparison. We assume that there are a total of 800 large packets to be transmitted. The channel gain is randomly distributed within [3, 5] dB in the first 200 large packet transmission, and randomly distributed within [4, 11] dB in the next 600 large packet transmissions. Since the PUBACK, PUBREC, PUBREL, and PUBCOMP packets only contain 2 bytes of fixed and variable headers, the packet sizes $U_{ACK}$, $U_{REC}$, $U_{REL}$, and $U_{COMP}$ are set as 0.032 kbits. The setting of other simulation parameters is summarized in Table 1.

![Figure 3. The weighted sum of packet-loss ratio and delay versus the number of large packet transmission.](image-url)

Table 1. Simulation parameters.

| Parameter                                      | Value       |
|------------------------------------------------|-------------|
| The total number of large packets $I$          | 800         |
| The total number of small packets $J$          | 10          |
| Channel bandwidth $B$                         | 0.2 MHz     |
| The size of each small packet $U(i)$          | [1, 2] kbits|
| Signal-to-noise ratio threshold $Q_{th}$       | 0.16        |
| The weight $V$                                | 0.9         |
| The transmission power $P$                    | 35 mW       |
| The feedback transmission power $P_{\text{back}}$ | 30 mW      |
| The data deduplication delay $t_{c}$           | 10 ms       |
| The maximum waiting time $t_{0}$              | 5 ms        |
| The weight of exploration $\omega$            | 2           |

Figure 3 shows the weighted sum of packet-loss ratio and delay versus the number of large packet transmission. Simulation result shows that after 200 large packet transmissions, all the curves show the downward trend, and the performance of QoS0 decreases the fastest. The reason is that the packet-loss ratio of QoS0 decreases due to the channel gain improvement after 200 large packet transmissions, while QoS1 and QoS2 are less affected by the channel gain based on the retransmission mechanism. DR-MQLS outperforms the single and fixed QoS level selection strategies of QoS0, QoS1, and QoS2 in weighted sum of packet-loss ratio and delay by 16.06%, 24.46%, and 44.86%, respectively. The reason is that DR-MQLS can dynamically select the optimal QoS level through trading off the exploration and exploitation based on the empirical performance.

Table 2 shows the delay versus the number of large packet transmission. Simulation result demonstrates that the delay of DR-MQLS is slightly higher than QoS0. The reason is that there is no retransmission mechanism and deduplication process in QoS0 level. It performs best in terms of delay, but sacrifices the...
packet-loss ratio as shown in Figure 3. When i = 800, compared with QoS1 and QoS2, the delay of DR-MQLS is decreased by 29.79% and 47.62%, respectively.

Figure 4 shows the optimal QoS level selection probability versus the number of large packet transmission. The optimal QoS level selection probability of DR-MQLS converges to 60.10% when the number of large packet transmission reaches 200. After 200 QoS level selections, the optimal QoS level selection probability first decreases and then reconverges to 87.77%. The reason is that DR-MQLS needs to relearn the QoS level selection strategy due to the significant change of channel state. After 200 large packet transmissions, compared with random QoS level selection, DR-MQLS has a significant advantage in the optimal QoS level selection probability due to the interaction and learning ability with the dynamic environment.

Figure 5 shows the weighted sum of packet-loss ratio and delay versus G_{th}. Simulation result shows that as the threshold increases, the delay shows a downward trend, while the packet-loss ratio shows the opposite trend. The reason is that as V increases, DR-MQLS lays more emphasis on delay minimization rather than packet-loss ratio reduction. DR-MQLS can dynamically balance the trade-off between packet-loss ratio and delay by adjusting the value of V, so as to satisfy the differentiated QoS requirements of EIoT services. The simulation results can provide a reference for the setting of the weight V in practical applications.

Figure 6 shows the weighted sum of packet-loss ratio and delay versus G_{th}.

Figure 7 shows the impact of V. Simulation result shows that as V increases, the delay shows a downward trend, while the packet-loss ratio shows the opposite trend. The reason is that as V increases, DR-MQLS lays more emphasis on delay minimization rather than packet-loss ratio reduction. DR-MQLS can dynamically balance the trade-off between packet-loss ratio and delay by adjusting the value of V, so as to satisfy the differentiated QoS requirements of EIoT services. The simulation results can provide a reference for the setting of the weight V in practical applications.

### Table 2. Average delay versus the number of large packet transmission.

| Algorithm | i = 200 | i = 400 | i = 600 | i = 800 |
|-----------|---------|---------|---------|---------|
| QoS0      | 0.41 s  | 0.32 s  | 0.30 s  | 0.28 s  |
| QoS1      | 0.59 s  | 0.51 s  | 0.48 s  | 0.47 s  |
| QoS2      | 0.76 s  | 0.68 s  | 0.65 s  | 0.63 s  |
| DR-MQLS   | 0.58 s  | 0.43 s  | 0.36 s  | 0.33 s  |

DR-MQLS: delay-reliability-aware MQTT QoS level selection.
Conclusion
In this paper, aiming at the QoS guarantee problem for EIoT based on MQTT protocol, we proposed a UCB-based delay-reliability-aware MQTT QoS level selection algorithm named DR-MQLS to minimize the weighted sum of packet-loss ratio and delay under incomplete information. Compared with the single and fixed QoS level selection strategies, that is, QoS0, QoS1, and QoS2, DR-MQLS can reduce the weighted sum of packet-loss ratio and delay by 16.06%, 24.46%, and 44.86%. In addition, the optimal QoS level selection probability can converge to 87.77%. For the future investigation, the joint optimization of bandwidth allocation and QoS level selection will be considered to facilitate low transmission delay, reliable data transmission, and high throughput in EIoT.

Declaration of conflicting interests
The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding
The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This study was financially supported by the Science and Technology Project of State Grid Corporation of China under grant number 52094021N010 (5400-202199534A-0-5-ZN).

ORCID iDs
Hao Zhang https://orcid.org/0000-0003-3245-8836
Zhao Wang https://orcid.org/0000-0003-3536-6291

References
1. Zheng R, Wang HW and Zhao JZ. A unified management framework for EIoT systems based on metadata and event detection. IEEE Access 2019; 7: 112629–112638.
2. Cheena S and Naveen G. Communication protocol stack for constrained IoT systems. In: 2018 3rd international conference on internet of things: smart innovation and usages (IoT-SIU), Bhimtal, India, 23–24 February 2018. New York: IEEE.
3. Shin I, Choi D and Choi H. Monitoring of digital substations using DDS. In: 2019 54th international universities power engineering conference (UPEC), Bucharest, Romania, 3–6 September 2019. New York: IEEE.
4. Garcia-Carrillo D and Marin-Lopez R. Multihop bootstrapping with EAP through CoAP intermediaries for IoT. IEEE Internet Things J 2018; 5(5): 4003–4017.
5. Nikolov N. Research of MQTT, CoAP, HTTP and XMPP IoT communication protocols for embedded systems. In: 2020 XXIX international scientific conference electronics (ET), Sozopol, Bulgaria, 16–18 September 2020. New York: IEEE.
6. Oussman S, Ibrahima N and Claude L. Lightweight security scheme for MQTT/MQTT-SN protocol. In: 2019 Sixth international conference on internet of things: systems, management and security (IOTSMS), Granada, 22–25 October 2019. New York: IEEE.
7. Wang XY, Umehira M, Han B, et al. An efficient privacy preserving spectrum sharing framework for internet of things. IEEE Access 2018; 8(1): 34675–34685.
8. Zhou H, Wang XY, Liu Z, et al. Resource allocation for SVC streaming over cooperative vehicular networks. IEEE Trans Veh Technol 2018; 67(9): 7924–7936.
9. Davide B, Walter C, Federico T, et al. From IoT to cloud: applications and performance of the MQTT protocol. In: 2020 22nd international conference on transparent optical networks (ICTON), Bari, 19–23 July 2020. New York: IEEE.
10. Naeem F, Srivastava G and Tariq M. A software defined network based fuzzy normalized neural adaptive multi-path congestion control for the internet of things. IEEE Trans Netw Sci Eng 2020; 7(4): 2155–2164.
11. Chen L, Dong XZ, Kuang XY, et al. Towards ubiquitous power distribution communication: multi-service access and QoS guarantees for IoT applications in smart grid. In: 2019 IEEE innovative smart grid technologies—Asia (ISGT Asia), Chengdu, China, 21–24 May 2019. New York: IEEE.
12. Rolando H. Dynamic CoAP mode control in real time wireless IoT networks. IEEE Internet Things J 2019; 6(1): 801–807.
13. Zhou H, Ji YS, Wang XY, et al. EICIC configuration algorithm with service scalability in heterogeneous cellular networks. IEEE ACM Trans Netw Sci Eng 2017; 25(1): 520–535.
14. Naeem F, Tariq M and Poor HV. SDN-enabled energy-efficient routing optimization framework for industrial internet of things. IEEE Trans Ind Inform 2021; 17(8): 5660–5667.
15. Li XW, Zheng YK and Alshehr MD. Cognitive AmBC-NOMA IoV-MTS networks with IQI: reliability and security analysis. IEEE Trans Intell Transp Syst 2021; 24(1): 1–12.
16. Fourat H, Wang CX, Ai B, et al. Spectral/energy efficiency tradeoff of cellular systems with mobile femtocell deployment. IEEE Trans Veh Technol 2016; 65(5): 3389–3400.
17. Li XW, Li JJ, Liu YW, et al. Residual transceiver hardware impairments on cooperative NOMA networks. IEEE Trans Wireless Commun 2020; 19(1): 680–695.
18. Li XW, Zhao ML, Zeng M, et al. Impaired ambient backscatter NOMA system: reliability and security. IEEE Trans Commun 2021; 69(4): 2723–2736.
19. Sadeq AS, Hassan R, Alrawi SS, et al. A QoS approach for internet of things (IoT) environment using MQTT protocol. In: 2019 international conference on cybersecurity (ICoCSec), Negeri Sembilan, Malaysia, 25–26 September 2019. New York: IEEE.
20. Niruntasukrat A, Issariyapat C, Pongpaibool P, et al. Authorization mechanism for MQTT-based internet of things. In: 2016 IEEE international conference on communications workshops (ICC), Kuala Lumpur, Malaysia, 23–27 May 2016. New York: IEEE.

21. Lee S, Kim H, Hong DK, et al. Correlation analysis of MQTT loss and delay according to QoS level. In: The international conference on information networking 2013 (ICOIN), Bangkok, Thailand, 28–30 January 2013. New York: IEEE.

22. Nurwarsito H and Nugroho H. Implementation of smart traffic light prototype using MQTT protocol for emergency vehicles. In: 2021 8th international conference on computer and communication engineering (ICCCE), Kuala Lumpur, Malaysia, 22–23 June 2021. New York: IEEE.

23. Weerasinghe YSP, Maduranga MWP and Dissanayake MB. RSSI and feed forward neural network (FFNN) based indoor localization in WSN. In: 2019 national information technology conference (NITC), Colombo, Sri Lanka, 8–10 October 2019. New York: IEEE.

24. Ahmadon MAB and Yamaguchi S. Process-based anomaly detection and analysis for cyber-physical system with MQTT protocol. In: 2020 IEEE international conference on consumer electronics (ICCE), Las Vegas, USA, 4–6 January 2020. New York: IEEE.

25. Li XW, Huang MY, Liu YW, et al. I/Q imbalance aware nonlinear wireless-powered relaying of BSG networks: Security and reliability analysis. IEEE Trans Netw Sci Eng 2021; 8(4): 2995–3008.

26. Zhao L, Zhao WL and Hawbani A. Novel online sequential learning-based adaptive routing for edge software-defined vehicular networks. IEEE Trans Wireless Commun 2020; 20(5): 2991–3004.

27. Wang XY, Jiang FZ, Zhong L, et al. Intelligent post-disaster networking by exploiting crowd big data. IEEE Netw 2020; 34(4): 49–55.

28. Tariq M, Ali M, Naem F, et al. Vulnerability assessment of 6G-enabled smart grid cyber-physical systems. IEEE Internet Things J 2021; 8(7): 5468–5475.

29. Zhou ZY, Chen XY, Liao HJ, et al. Context-aware learning-based resource allocation for ubiquitous power IoT. IEEE Internet Things Mag 2020; 20(5): 2991–3004.

30. Modi N, Mary P and Moy C. QoS driven channel selection algorithm for cognitive radio network: multi-user multi-armed bandit approach. IEEE Trans Cognitive Commun Netw 2017; 3(1): 49–66.

31. Liu Q, Zhu Y and Liu G. Integrating the power distribution terminals into the power distribution internet of things. In: 2021 power system and green energy conference (PSGEC), Shanghai, China, 20–22 August 2021. New York: IEEE.

32. Evgenii M, Mikhail B and Vitaly K. Load distribution cluster in distributed IoT monitoring system for hydrographic network. In: 2020 international conference on industrial engineering, applications and manufacturing (ICIEAM), Sochi, Russia, 18–22 May 2020. New York: IEEE.

33. Liao HJ, Zhou ZY, Zhao XW, et al. Learning-based context-aware resource allocation for edge-computing-empowered industrial IoT. IEEE Internet Things J 2020; 7(5): 4260–4277.

34. Nikoiaos P and Theodore A. Resource allocation management for indoor power-line communications systems. IEEE Trans Power Deliv 2007; 22(2): 893–903.

35. Goswami P, Mukherjee A, Maity M, et al. A neural-network-based optimal resource allocation method for secure IoT network. IEEE Internet Things J 2022; 9(4): 2538–2544.

36. Kirubakaran R, Prathibhan C and Karthika C. A cloud based model for deduplication of large data. In: 2015 IEEE international conference on engineering and technology (ICETECH), Coimbatore, India, 20 September 2015. New York: IEEE.

37. Xu X and Tu Q. Data deduplication mechanism for cloud storage systems. In: 2015 international conference on cyber-enabled distributed computing and knowledge discovery, Xi’an, China, 17–19 September 2015. New York: IEEE.

38. Liao HJ, Zhou ZY, Kong WX, et al. Learning-based intent-aware task offloading for air-ground integrated vehicular edge computing. IEEE Trans Intell Transp Syst 2021; 22(8): 5127–5139.

39. Wang S and Zeng G. A novel approach to select high-reward data items in big data stream based on multi-armed bandit. IEEE Trans Comput Social Syst 2021; 1(1): 1–10.

40. Zhou H, Wang XY, Umehira M, et al. Wireless access control in edge-aided disaster response: a deep reinforcement learning-based approach. IEEE Access 2021; 9: 46600–46611.

41. Zhou ZY, Wang Z, Yu HJ, et al. Learning-based URLLC-aware task offloading for internet of health things. IEEE J Sel Areas Commun 2021; 39(2): 396–410.

42. Zhao L, Yang KQ, Tan ZY, et al. A novel cost optimization strategy for SDN-enabled UAV-assisted vehicular computation offloading. IEEE Trans Intell Transp Syst 2020; 22(6): 3664–3674.

43. Zhao L, Li HX, Lin N, et al. Intelligent content caching strategy in autonomous driving toward 6G. IEEE Trans Intell Transp Syst 2021; 1(1): 1–11.

44. Zhou ZY, Liao HJ, Zhao XW, et al. Reliable task offloading for vehicular fog computing under information asymmetry and information uncertainty. IEEE Trans Veh Technol 2019; 68(9): 8322–8335.

45. Yesenia G, Geovanni N, Gustavo S, et al. Comparative performance analysis between MQTT and CoAP protocols for IoT with raspberry pi 3 in IEEE 802.11 environments. In: 2020 15th Iberian conference on information systems and technologies (CISTI), Seville, 24–27 June 2020. New York: IEEE.

46. Zhang SD, You XH, Zhang PP, et al. A UCB-based dynamic CoAP mode selection algorithm in distribution IoT. Alexandria Eng J 2022; 61(1): 719–727.