sensors

MDPI

Article

Q-LBR: Q-Learning Based Load Balancing Routing for UAV-Assisted VANET

Bong-Soo Roh [1,2], Myoung-Hun Han [1], Jae-Hyun Ham [1] and Ki-Il Kim [2,*]

[1] Agency for Defense Development, Daejeon 34186, Korea; saintroh@add.re.kr (B.-S.R.); mengddor@add.re.kr (M.-H.H.); mjhham@add.re.kr (J.-H.H.)

[2] Department of Computer Science and Engineering, Chungnam National University, Daejeon 34134, Korea

* Correspondence: kikim@cnu.ac.kr; Tel.: +82-42-821-6856

Received: 3 September 2020; Accepted: 30 September 2020; Published: 5 October 2020

Abstract: Although various unmanned aerial vehicle (UAV)-assisted routing protocols have been proposed for vehicular ad hoc networks, few studies have investigated load balancing algorithms to accommodate future traffic growth and deal with complex dynamic network environments simultaneously. In particular, owing to the extended coverage and clear line-of-sight relay link on a UAV relay node (URN), the possibility of a bottleneck link is high. To prevent problems caused by traffic congestion, we propose Q-learning based load balancing routing (Q-LBR) through a combination of three key techniques, namely, a low-overhead technique for estimating the network load through the queue status obtained from each ground vehicular node by the URN, a load balancing scheme based on Q-learning and a reward control function for rapid convergence of Q-learning. Through diverse simulations, we demonstrate that Q-LBR improves the packet delivery ratio, network utilization and latency by more than 8, 28 and 30%, respectively, compared to the existing protocol.

Keywords: MANET; VANET; UAV relay; load balancing; routing; Q learning

1. Introduction

The vehicular ad hoc network (VANET), a special type of mobile ad hoc network (MANET), has been investigated to provide the infrastructure of a new service paradigm through self-organizing networks that exist between vehicles. However, it still experiences difficulty in routing with easily disconnected features that are associated with dynamic wireless environments in mobile network topologies. To overcome this problem, the deployment of unmanned aerial vehicles (UAVs) via the cooperation of vehicles has been considered.

Several methods have recently been developed in the literature for UAV-assisted network protocols that address the issues of high mobility in the network and unpredictable change in topology of the mobile nodes [1–6]. Unlike a fixed ground relay station, a UAV relay node (URN) moves along with the ground vehicular nodes (GVNs) to support a reliable network through a continuous line-of-sight (LoS) link. In addition, considering the characteristics through which MANET is temporarily constructed and operated, this is an extremely economical solution compared to the construction of a ground infrastructure. In the case of a VANET, in particular, the relay node is faced with the risk of a broken link that can be caused by mobility, and nonline-of-sight (NLoS) can occur more frequently than in a general MANET. Therefore, a UAV-assisted relay can be a more useful tool when operating in a VANET environment. Because the UAV relay path is most likely to be the best approach in terms of link quality and the number of hops, it is highly likely that a bottleneck of the URN will occur from the existing routing protocol when the network is congested. This bottleneck can degrade the transmission efficiency of the UAV and, in the case of Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA

MAC, Ex. 802.11p), the channel access opportunities of all the ground nodes can be lost. In addition, the spatial frequency reusability of the ground node can be lowered, resulting in a decrease in the overall network performance. However, if a UAV relay node (URN) is operated only as a backup path when a ground link disconnection occurs, the URN resources are wasted. Therefore, to increase the efficiency of a URN, it is necessary to design a structure that can handle the maximum acceptable traffic while maintaining a certain level of ground network load.

In the design of the Q-learning based load balancing routing (Q-LBR), a URN uses an overhearing technique based on the broadcast nature of wireless media to recognize the ground network load in the message transmitted between the GVNs. The URN then distributes the UAV routing policy area (URPA) information based on the Q-learning method through broadcast messages such as Hello. The GVNs decide whether to use the air relay path through the received URPA and the current UAV relay load, and the URN continuously maximizes the network throughput even when the wireless environment is dynamically changing. We propose a Q-LBR system for load balancing in a UAV-assisted VANET. This provides a method for URN to handle the maximum traffic acceptable while maintaining a certain level of ground network load.

The main contributions of this study are fourfold:

(1) We propose a low-overhead technique for estimating the network load through the queue status obtained by the URN from each GVN. To estimate a network load with low overhead, we design a technique using overhearing and broadcast messages received from the GVNs. This is possible because the URN can cover an area wider than that of the GVNs.
(2) We propose a load balancing scheme based on Q-learning, which can enable dynamic network load control within the usable capacity of the URN. Q-LBR defines the URPA when considering the traffic characteristics and the existence of a route independently from the ground network routing.
(3) We propose a reward control function (RCF) to enable rapid learning feedback of the reward values in consideration of a dynamic network environment. Q-LBR adjusts the reward value based on the URN load and ground network congestion.
(4) We implemented the Q-LBR on a network-level simulator using the Riverbed Modeler (formerly OPNET) and experimentally evaluated its performance. Our evaluation results showed that Q-LBR achieved a significantly better packet delivery ratio (PDR) and network utilization and latency according to the traffic load conditions than existing algorithms and Q-LBR without Q-learning.

2. Related Studies

This section describes the UAV-aided routing and load balancing routing as well as the Q-learning-based routing, which have been designed to enhance the existing VANETs. Their respective limitations are then addressed to clarify our motivation for the Q-LBR design.

2.1. UAV-Assisted Routing Protocols

Research on various applications using drones has been rapidly increasing, and that on UAV-assisted VANETs is also actively underway. Load carry and deliver routing (LCAD) [3] was proposed for static single-hop routing for UAVs to assist ground nodes. LCAD provides a load-carry-delivery mechanism for enhancing connectivity during the data delivery process of a sparsely connected network by applying the disruption tolerant network (DTN) concept. As a drawback, LCAD does not consider the traffic characteristics of a URN bottleneck. UAV-assisted VANET routing (UVAR) [4] and its extension (U2RV, UAV-assisted reactive routing protocol for VANETs) [5] utilize reactive multipath routing for UAV-assisted routing. UVAR and U2RV are based on four main processes: discovery, selection, data delivery and maintenance. These protocols calculate a multi-criteria score for considering the highest degree of connectivity and the shortest distance, as well as the minimum delay to the target destination. In terms of the selection process, the score is calculated for every discovered path by combining several metrics. As a result, the best scored

path will be selected, and it is difficult to guarantee the service Quality of service (QoS) when the traffic is concentrated on the corresponding path. Therefore, similar to LCAD, these protocols do not consider the traffic characteristics or URN bottlenecks. A multi-UAV-aided MEC architecture has been proposed [6] as a joint multi-UAV deployment and task scheduling optimization for IoT networks. This paper proposed the task scheduling method using deep reinforcement learning in terms of the role of the Multi-access Edge Computing (MEC) node. However, in terms of the relay node of the URN, routing and load balancing for the traffic priority and characteristics were not considered.

2.2. Load-Balancing Routing Protocols

With the rapid increase in the use of VANETs, and increased demand of networked vehicles for a wide range of services and better information, load balancing has become an essential and important research area. Efficient load balancing ensures efficient resource utilization and enhances the overall performance of the network system. UAV-aided cross-layer routing (UCLR) [7] is a cross-layer routing and load-balancing algorithm that considers the UAV relay based on Open Shorstest Path First-MANET Designed Router (OSPF-MDR). The routing metric of UCLR is calculated using the packet error rate (PER), and load balancing is adjusted using a static threshold of the queue length. Although UCLR handles dynamic UAV traffic load issues between the URN and GVNs, the main drawback of the UCLR is its static load control with a dynamic network environment. In a UAV-assisted VANET environment there are several moving GVNs, and the changes in the traffic patterns are also extremely rapid. Therefore, there is a need for a dynamic load control scheme capable of responding to rapidly changing network environments. Moreover, UCLR does not consider a method for improving the utilization of the UAV throughput. A hierarchical routing scheme with load balancing (HRLB) has been proposed [8] as a hierarchical geography routing protocol for software-defined VANETs. HRLB constructs a path cost function with load balancing and maintains two paths with minimal costs from the selected grids. This protocol considers the load only from the GVN and disregards the UAV-assisted relay. A queue utilization routing algorithm (QURA) has been proposed [9] as a machine learning-based routing scheme for QoS routing. This protocol applies an artificial neural network (ANN) to routing and selects the next hop according to a queue utilization prediction (QUP). However, supervised learning has a problem in that it is difficult to create the training data and the label in dynamic network topology. Table 1 summarizes the characteristics of the UAV-assisted routing and load balancing protocols. By reviewing these protocols, we can state that most of the proposed routing protocol techniques designed for UAV-assisted VANET disregarded the traffic characteristics and dynamic load balancing in congested network environments. In addition, these routing protocols do not consider traffic bottlenecks owing to a better link quality and hop count compared to the ground network of a UAV relay node.

Table 1. Comparative study between routing protocols.

Features	LCAD [1]	U2RV [2]	UCLR [3]	HRLB [4]	Q-LBR [5]
Multipath	No	Yes	Yes	Yes	Yes
UAV-assisted Relay	Yes	Yes	Yes	No	Yes
Traffic Characteristics	No	No	No	No	Yes
Load Balancing	No	No	Yes	Yes	Yes
Dynamic Load Control	No	No	No	No	Yes
Machine Learning	No	No	No	No	Yes
Type of network	UAV/MANET	UAV/VANET	UAV/MANET	VANET	UAV/VANET

[1] LCAD: Load Carry and Deliver Routing. [2] U2RV: UAV-assisted Reactive Routing Protocol for VANET. [3] UCLR: UAV-aided Cross-Layer Routing. [4] HRLB: Hierarchical Routing Scheme with Load Balancing. [5] Q-LBR: Q-learning based Load Balancing Routing.

To address the aforementioned problems, we propose a new load-balancing routing scheme that is capable of achieving efficient operation of UAV relay nodes in consideration of the traffic characteristics.

In addition, we use the Q-learning algorithm, which improves the convergence speed of the reward function for dynamic load control.

2.3. Q-Learning-Based Routing Protocols

In recent years, artificial intelligence techniques, which include machine learning, have attracted a significant amount of interest from researchers of various fields [8]. Among such techniques, reinforcement learning (RL) is being investigated in wireless systems because it provides a solution to optimize the system parameters by learning the surrounding area in a dynamic and complicated wireless environment [10–12]. Q-learning is a representative RL, and studies on using this approach to allocate routing policies in a dynamically changing network environment have been conducted. The Q-learning algorithm [13] solves this problem by utilizing the following Q-value update equation:

$$Q(s_{t+1}, \alpha_{t+1}) \leftarrow (1-\alpha)Q(s_t, \alpha_t) + \alpha\{f_r(s_t, \alpha_t) + \gamma max_{\alpha'}(Q(s_t, \alpha_t), \alpha')\}, \quad (1)$$

where $Q(s_t, \alpha_t)$ is the Q-value of the current state s_t when action α is selected at time t, $f_r(s_t, \alpha_t)$ represents the reward function when state s_t selects action α_t, and max$(Q(s_t, \alpha_t), \alpha')$ is the maximum possible Q-value in the next state s_{t+1} when possible action α' is selected. The learning rate α and discount factor γ have values between zero and one. As an advantage of Q-learning, it can be used to design optimal policy functions even in unknown environments. In general, a wireless network environment is extremely complex and difficult to predict and, therefore, it is considered that reinforcement learning such as Q-learning is more suitable than supervised learning.

There are several noteworthy studies on Q-learning-based routing protocols. Q-Geo [14] proposed an ad hoc routing method based on geographic information through Q-learning in an unmanned robotic network. This algorithm enables network enhancement using local information without full network knowledge by calculating the packet travel speed. The energy-aware QoS routing protocol (EQR-RL) [15] uses a reinforcement learning algorithm and the reinforcement learning based geographic routing (RLGR) [16] are proposed methods for applying Q-learning in routing decisions for a network lifetime enhancement in a Wireless sensor network (WSN). Q-learning based fuzzy logic [17] for multi-objective routing algorithm is proposed as a method for flying ad hoc networks (FANET).

Although there have been numerous studies applying Q-learning, results for UAV-assisted VANET are yet to be presented. In addition, the key issue for applying RL to a rapidly changing network environment is solving the convergence speed problem. Specifically, RL is based on the results of experiences acquired through exploration and, thus, it sometimes takes significant trial and error to obtain meaningful results. Likewise, until recently, reinforcement learning in the field of networking has not been considered.

3. System Model and Assumptions

In this section, we describe the system model and some key network assumptions. Q-LBR assumes that the UAV relay node has a low and constant altitude during flight to be able to relay with vehicles on the ground, and that all network nodes have the same RF performance. However, URN has a relatively low signal attenuation owing to high altitude compared to the ground node. Therefore, a URN can provide superior performance in terms of radio coverage and link quality.

Consider a circular geographical area of radius r as depicted in Figure 1 in which a UAV is deployed to provide wireless coverage for ground users located within the area. For air-to-ground channel modeling, a common approach is to consider the LoS and NLOS links between the UAV and the ground users separately [18]. The coverage probability (P_{cov} [19]) for the ground node, located at a distance $r \leq r_u = htan\left(\frac{\theta_B}{2}\right)$ from the projection given UAV_j in the area, is provided by Equation (2):

$$P_{cov} = P_{LoS,j}\left(\frac{P_{min} + L_{dB} - P_t - G_{3dB} + \mu_{LoS}}{\sigma_{LoS}}\right) + P_{NLoS,j}(\frac{P_{min} + L_{dB} - P_t - G_{3dB} + \mu_{NLoS})}{\sigma_{NLoS}}, \quad (2)$$

where $P_{min} = 10\log(\beta N)$ is the minimum received power requirement for a successful detection, N is the noise power, and β is the signal-to-noise ratio (SNR) threshold. In addition, L_{dB} is the path loss, and G_{3dB} is the antenna gain ($G_{3dB} \approx 29000/\theta_B^2$).

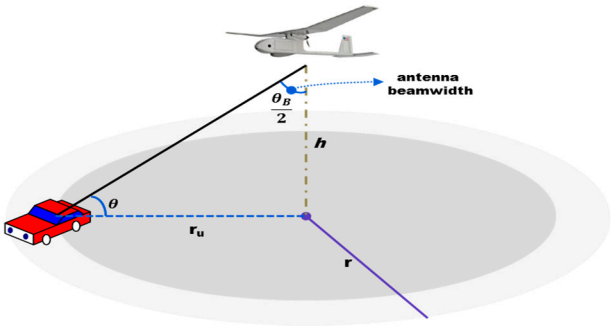


Figure 1. UAV relay coverage model.

Because 802.11p is expected to be widely used in industrial areas, and is the most suitable for VANET [20–23], we adopted the IEEE 802.11p MAC protocol for both inter-GVN communication and UAV-to-GVN communication.

We classified the following three types according to the characteristics of the data services based on the packet priority for an efficient operation of a URN in a congested network environment.

(1) Urgent service message (USM): Highest priority services that need to be urgently sent.

(2) Real time service (RTS): Medium-priority services with delay constraints but little packet loss.

(3) Connection oriented protocol (COP): Lowest priority services with less sensitivity to delay and loss.

In terms of network services, it is extremely important to select a routing path by considering traffic characteristics. From the user's perspective, the effects experienced by a packet loss or delay are extremely different depending on the traffic characteristics. For example, there is a considerable difference between a streaming service that requires real-time and delay-insensitive TCP services.

4. Proposed Q-LBR Design

In this section, we describe the Q-LBR design in detail. Q-LBR is designed to maximize the network utilization of a URN through load balancing. Q-LBR introduces new mechanisms in UAV-assisted VANET, as described in Figure 2.

The Q-LBR protocol consists of two phases, as described in Figure 3. During the first phase, a URN collects a ground network congestion identifier (GNCI) to the GN messages through broadcast and unicast overhearing to determine the congestion level of the ground network. Through this phase, the URN can recognize the congestion level of the ground network based on the collected GNCI and UAV relay congestion identifier (URCI) information. During the second phase, the URN disseminates URPA information corresponding to the action of the Q-learning. Specifically, the URN substitutes the GNCI and URCI into the Q-learning states and feeds the appropriate reward value back based on an RCF calculation. Finally, the result of the RCF determines the URPA value, which is divided into upper and lower values, and shares it with a Hello message.

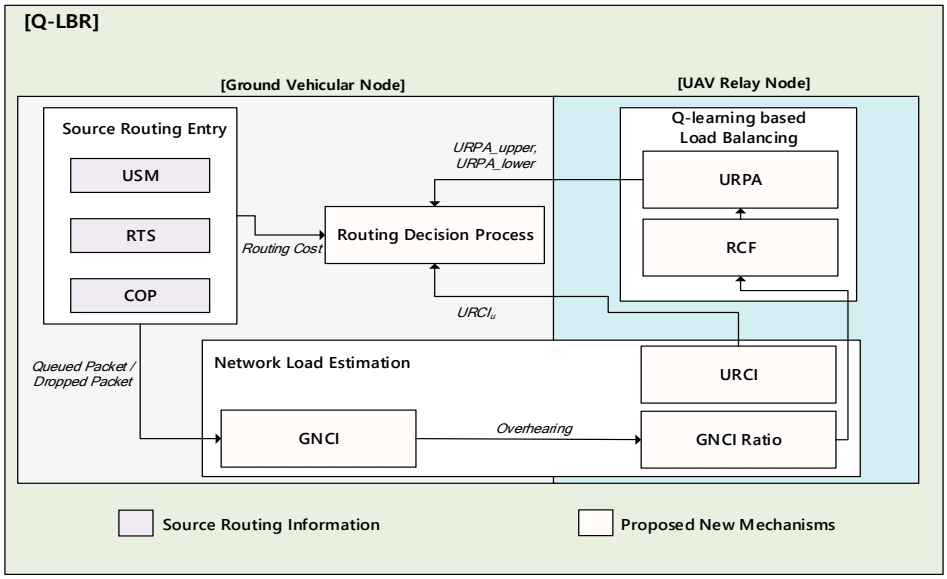


Figure 2. Q-learning based load balancing (Q-LBR) framework.

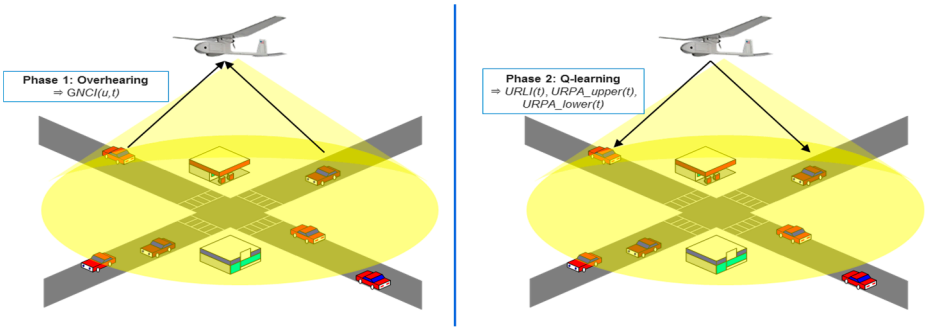


Figure 3. Two phases of Q-LBR.

4.1. Path Discovery and Maintenance

The path discovery of Q-LBR is performed by route request (RREQ) flooding, and the basic routing search method is similar to the source-based multipath routing protocol adopted in the existing VANET. The destination node responds to the RREQ, including the optimal and suboptimal paths, using a route reply (RREP) message. This increases survivability of the VANET routing through the use of suboptimal paths when the optimal path is disconnected. In the path-discovery process, the URN can receive multiple RREQs for the same destination from many GVNs, and thus the number of URN responses is limited. Through the Q-LBR path discovery process, the source node can acquire route information to the destination node, including a URN. Q-LBR periodically transmits a probe packet for routing updates to maintain the optimal and suboptimal paths. If all paths are disconnected, the intermediate node sends a route error (RERR) message to the source node.

4.2. Network Load Estimation

4.2.1. Ground Network Congestion Identifier

It is extremely important to determine how a URN identifies ground network congestion according to the traffic load. In brief, each GVN estimates the GNCI from itself by using the queue load. This bitwise information is delivered to the URN using overhearing or broadcast messages. Then, URN computes the ratio of GNCI ($GNCI_{ratio}$) in its time interval by using the number of $GNCI_i$ instances with a value of '1' from GVN i.

For a more detailed explanation, $q_{ground_i}(t)$, given by Equation (3), which indicates the queue load of each GVN, is calculated as the ratio of the maximum queue length (MQL_i) to the average queue length ($AQL_i(t)$) corresponding to time t of GVN i.

$$q_{ground_i}(t) = \frac{AQL_i(t)}{MQL_i} \tag{3}$$

Based on the result of $q_{ground_i}(t)$, each GVN calculates the weighted moving average $Q_{ground_{i,k}}(t)$, given by Equation (4), in the window size k from GVN i.

$$Q_{ground_{i,k}}(t) = \frac{\sum_{k=0}^{n} w_k q_{ground_i}(t-k)}{\sum_{k=0}^{n} w_k} \tag{4}$$

Each ground node i determines whether the result of $Q_{ground_i}(t)$ exceeds the GVN load threshold $Q_{ground_{th}}$, given by Equation (5), and marks the value of $GNCI_i(t)$ with a '1' or '0' in the packet header.

$$GNCI_i(t) = \begin{cases} 1 \; if \; Q_{ground_i}(t) > Q_{ground_{th}} \\ 0 \quad otherwise \end{cases} \tag{5}$$

The URN receives $GNCI_i(t)$ of each ground node through an overhearing or broadcast messages and then calculates $GNCI_{ratio}(t)$, given by Equation (6), which is the ratio of the congested GVN to the total number of GVNs, N.

$$GNCI_{ratio}(t) = \frac{\sum_{i=0}^{N} GNCI_i(t)}{N(t)} \tag{6}$$

4.2.2. UAV Relay Congestion Identification

The URCI, given by Equation (7), is calculated through the URN's own queue load from the UAV relay node u.

$$URCI_u(t) = \frac{AQL_u(t)}{MQL_u} \tag{7}$$

With $URCI_u$, however, it can be recognized that the closer $AQL_u(t)$ is to MQL_u, and when considering the load balancing aspect, the greater the throughput within the maximum range the UAV can accommodate.

4.3. Q-Learning-Based Load Balancing

4.3.1. Q-Learning Design for UAV-Assisted Network

Q-learning is a model-free reinforcement learning algorithm that finds an estimate of the optimal action-value function. It is able to compare the expected reward of the available actions for a given state without requiring a specific model of the network environment. Q-learning finds an optimal policy, in the sense that the expected value of the total reward return over all successive iterations is the maximum achievable. Figure 4 shows the Q-learning mechanism of the proposed method.

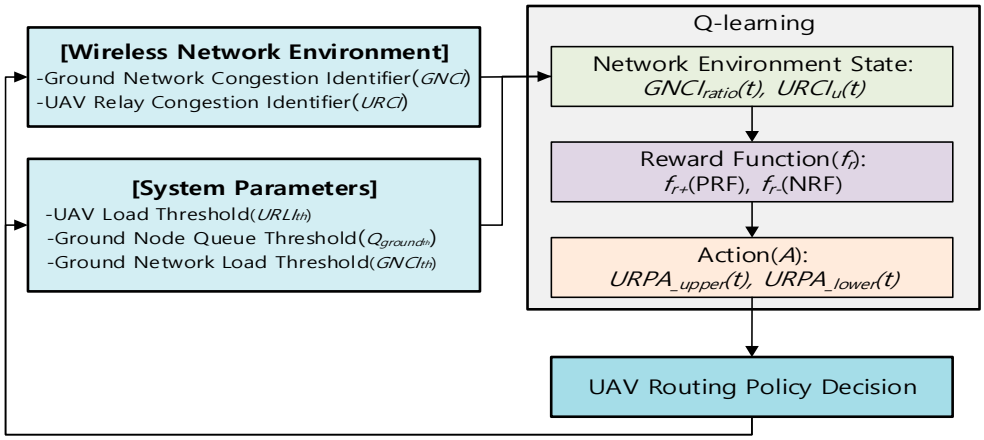


Figure 4. Q-learning design for Q-LBR.

An URN is an agent of Q-learning, and its action is a selection of URPA for the UAV routing policy decision. In Q-LBR, URN's experience consists of a sequence of episodes. In the N_{th} episode, when URN finds a $URPA\_upper$ and $URPA\_lower$ that satisfies $URCI_{th}$ and $GNCI_{th}$, learning is terminated. If a network change occurs, and the $URCI_{th}$ and $GNCI_{th}$ are not satisfied, the learning process is repeated. Specifically, according to Figure 5 and Algorithm 1, the URN recognizes the wireless network environment through the $GNCI_{ratio}$ and $URCI_u$, then the URN learns in the network based on Q-learning and provides an appropriate reward f_r according to $GNCI_{ratio}$ and $URCI_u$. The reward function f_r selects f_{r+}(PRF) in $URCI_u(t) \leq URCI_{th}$ situation and selects f_{r-}(NRF) otherwise.

States	Actions					
	URPA_upper			*URPA_lower*		
	f_{r+}	keep	f_{r-}	f_{r+}	keep	f_{r-}
$URCI_u(1)$, $GNCI_{ratio}(1)$						
…						
$URCI_u(N)$, $GNCI_{ratio}(N)$						

Figure 5. Q-table structure for Q-LBR.

To recognize the state of the ground network, the URN listens to $GNCI_i$ transmitted from GVN i using an overhearing or broadcast messages. At time t, the URN can calculate $GNCI_{ratio}$ from the total number of N nodes. At the same time, the URN can calculate $URCI_u$ from its own queue load. The learning goal of Q-LBR is to find an optimal URPA that is as close as possible to $URCI_{th}$, which indicates the allowable load of the URN and satisfies an appropriate level of ground network load $GNCI_{th}$. If the URN finds the optimal URPA, the URN maintains its current state until it changes into a new network state. If not, the URN updates the Q-table according to the Q-learning procedure such that the reward value by the URPA actions can be maximized. Finally, the results of $URPA\_upper$ and $URPA\_lower$ corresponding to the action of the Q-learning are distributed to the GVNs. Through a repetitive execution of this process, the URN can find the optimal policy for the URPA suitable for the network environment.

Algorithm 1: Q-learning based Load Balancing

URN ← UAV relay node;
GVN ← Ground vehicular node;
$GNCI_i$ ← Ground node congestion identifier from node i;
$GNCI_{ratio}$ ← Ratio of congested GVNs;
$URCI_u$ ← UAV relay congestion identifier from URN;
$URCI_{th}$ ← Threshold of URCI;
$URPA\_upper$ ← Upper boundary value of UAV routing policy area;
$URPA\_lower$ ← Lower boundary value of UAV routing policy area;
f_r ← Reward function

for $t \rightarrow 1$, n **do**
  for $i \rightarrow 1$, N **do**
    URN listens to $GNCI_i$ using overhearing or broadcast messages from node i
    URN calculates $GNCI_{ratio}$ at time t received from total number of N
    URN calculates $URCI_u$ at time t from its own queue load

    if ($GNCI_{ratio} < GNCI_{th}$ && $URCI_u \cong URCI_{th}$) **then**
      URN maintains its current state
    Else
      URN calculates the reward $f_r(t\text{-}1)$ for the previous action $a(t\text{-}1)$ at state s(t-1)
      URN updates the Q-value of ($s(t\text{-}1)$, $a(t\text{-}1)$) in Q-table
      URN determines the current state $s(t)$ based on the $GNCI_{ratio}$ and $URCI_u$
      URN selects the optimal action a(t) for the next t+1 time period
    end if

    URN distributes $URCI_u$ and URPA_upper and URPA_lower to GVN
  end for
end for

4.3.2. UAV Routing Policy Area

In a rapidly changing network environment, it is important to narrow and simplify the scope of the problem to be solved in order to design an optimal policy for an effective URN routing through the RL. If a learning algorithm is designed, including ground network routing, the problem to be solved becomes more complicated and the reward through the RL becomes difficult to effectively reflect. Therefore, Q-LBR defines URPA corresponding to two knobs ($URPA\_upper$ & $URPA\_lower$) when considering the priority of traffic and the existence of a route independently from the ground network routing.

URPA is a parameter for applying the URN routing policy, and is defined in the following three policy areas to determine whether or not to be the route of an air node relay when a URN is present on the routing path of the GN. URPA sets the boundary for the policy area based on the parameters of $URPA\_upper$ and $URPA\_lower$ ($URPA\_upper > URPA\_lower$), as shown in Figure 6, and dynamically changes with time t based on the action of the Q-learning.

- Policy Area A: Allow a UAV relay only when there is no ground path with a high-priority packet
- Policy Area B: Allow a UAV relay only when there is no ground path without considering the packet priority.
- Policy Area C: Allow a UAV relay without considering the packet priority or existence of the ground path (allow all traffic)

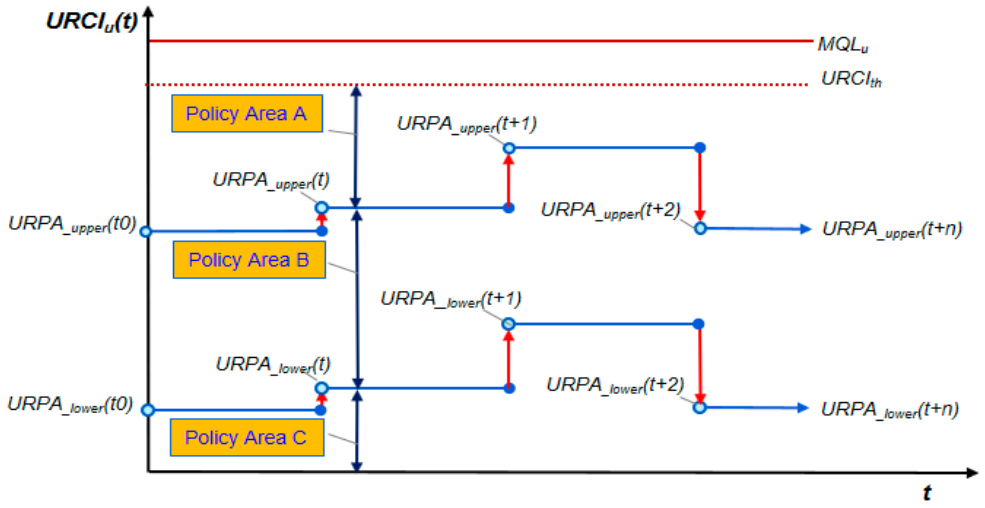


Figure 6. UAV routing policy area (URPA) example.

4.3.3. Reward Control Function Design for Rapid Convergence

Reinforcement learning is a problem faced by an agent who must learn behavior through trial-and-error in a dynamic environment. However, the learning method can cause a convergence speed problem in terms of the time required to find the optimal state. In particular, the network environment is changed by various variables over time, and thus a method allowing the reinforcement learning system to respond quickly is required. Previous studies in which Q-learning was applied were generally proposed to control the learning rate through the value of α. However, if α is too large, it is difficult to converge to the optimal value function and, if it is too small, it takes too long to learn. This shows that there is a limitation in coping with rapid changes in the network with the existing method through the reflection ratio of the learned results.

Q-LBR proposes using an RCF to determine the reward according to $URCI_u(t)$ and $GNCI_{ratio}(t)$ for the purpose of improving the convergence speed of the reward function. The RCF of Q-LBR dynamically determines the reward value according to the load-state of the URN and the ground network congestion with the rapidly changing network environment. Specifically, if the queue load of the URN is sufficient, a large positive reward value is given to increase the utilization of the URN. By contrast, under high congestion, a large negative value is given to reduce the URN and ground network congestion.

The reward function ($f_r(s_t,\ a_t)$), given by Equation (8), is as follows:

$$f_r(s_t,\ a_t) = \begin{cases} f_{r+}(s_t,\ a_t) & if\ URCI_u(t) \le URCI_{th} \\ f_{r-}(s_t,\ a_t) & else \end{cases} \tag{8}$$

The positive reward function (PRF), given by Equation (9), for action a is expressed as follows:

$$f_{r+}(s_t,\ a_t) = -1/ln(k * (1 - \lambda(t)), \tag{9}$$

where $\lambda(t)$, given by Equation (10), is a function ($\lambda(t) \in (0, 1]$) for determining the reward values according to $URCI_u(t)$ and $GNCI_{ratio}(t)$ (where $URCI_u(t) \le URCI_{th}, GNCI_{ratio} \le GNCI_{th}$). Here, k is the scale parameter (k > 0). When $\lambda(t)$ is high, the reward value is significantly increased. When the value of $\lambda(t)$ is low, it gradually increases.

$$\lambda(t) = w_1 * \left(\frac{URCI_u(t)}{URCI_{th}}\right) + w_2 * \left(\frac{GNCI_{ratio}(t)}{GNCI_{th}}\right) \tag{10}$$

The negative reward function (NRF) for action a is expressed as follows:

$$f_{r-}(s_t,\ a_t) = ln(k * (1/r_{max} - \lambda(t)), \tag{11}$$

where r_{max} is the maximum reward value ($r_{max} > \lambda(t),\ \ r_{max} > 0$). The NRF is also controlled by $\lambda(t)$ and the weight w of $URLI(t)$ and $GNCI(t)$. In contrast to the PRF, when $\lambda(t)$ is high, the reward value is significantly decreased, and when the value of $\lambda(t)$ is low, the reward value gradually decreases.

4.4. Routing Decision Process

According to Algorithm 2, the ground source node can receive p messages owing to multiple paths from the ground destination node. Through this message, routing metrics are calculated in $RREP_p$ packets for each routing path. If $RREP_p$ including an URN exists, and this path is less expensive than the ground path, the $URCI_u$ of the URN and the traffic priority (TP) of the packets check the URPA condition. If all the conditions are satisfied, the path including the URN can be selected as the optimal path. If unsatisfied, the next suboptimal ground path is selected.

Algorithm 2: Routing Decision Process

S ← Ground source node;
D ← Ground destination node;
URN ← UAV relay node;
RCU ← Routing cost including UAV path;
RCG ← Routing cost with GVN only;
URPA ← UAV routing policy area;
$URCI_u$ ← UAV relay congestion identifier from URN;
TP ← Traffic priority

for $p \rightarrow 1$, n **do**
  if S receives $RREP_p(D)$ packet **then**
    Calculate routing cost using metric information collected in $RREP_p(D)$ packet
    if ($RREP_k$ path contains URN || RCU < RCG) **then**
      if ($URCI_u$ and TP satisfy URPA's UAV relay conditions) **then**
        Select the routing path that includes the URN as the optimal route
      else
        Select the suboptimal ground path
      end if
    Else
      Select the optimal ground path
    end if
  end if
end for

5. Simulation Results and Analysis

5.1. Simulation Environments

In this section, we evaluate the performance of the proposed protocol using the network simulator Riverbed Modeler version 18.7. We summarize the detailed information regarding our simulation parameters in Table 2.

During the simulation, three types of packets are considered: USM, RTS, and COP packets. USM is a traffic type corresponding to the emergency data and control message of a critical service, and is set to EF, the highest packet priority. The size of the USM packet is set to 256 bytes based on an exponential distribution, and the packet interval is set to 10 requests per second (r/s).

Since the traffic size and request rate follow the exponential distribution f_X with parameter λ_s as follows:

$$f_X(x) = \lambda_s e^{-\lambda x} \tag{12}$$

RTS is a traffic type corresponding to a service requiring a certain amount of real-time data using a codec such as a video stream. The priority of the RTS packet is set to AF21, which is the middle priority of the packet. The size of the RTS packet is set to 1500 bytes, and the packet interval is set to 10 r/s. COP is a traffic type corresponding to TCP data, such as FTP, and is set to CS0, the lowest packet priority. The size of the COP packet is set to 256 bytes based on an exponential distribution the same as USM, and the packet interval is set to 10 r/s. To support the QoS requirements for different services, the IEEE 802.11p EDCA mechanism defines four access categories (AC0–AC3) for each channel. We defined AC0 through AC2 for mapping to USM, RTS, and COP services, respectively. The arbitration interframe space (AIFS) is determined according to the mapping relationship for each service. AIFS indicates the idle channel time that must be endured for a transmission opportunity.

Table 2. Simulation Parameters.

Layers	Parameters	Settings
PHY	Data Rate	1 Mbps
	Propagation Loss Model	Urban Propagation Model
	Coverage Probability (Air-to-Ground)	P_{cov} [13]
	Frequency Band	5.9 GHz
MAC	Protocol	802.11p
	Slot Time	13 µs
	SIFS	32 µs
	AIFNSN[AC0:USM/AC1:RTS/AC2:COP]	2, 3, 6
Network	Hello Interval	30 s
	Active Route Timeout	90 s
Application	USM Traffic (Size/Rate)	Exp. 256 bytes/Exp. 10 rps
	RTS Traffic (Size/Rate)	Con. 1500 bytes/Con. 10 rps
	COP Traffic (Size/Rate)	Exp. 256 bytes/Exp. 10 rps
Q-learning	Learning rate(α)	0.3
	Discount Factor(γ)	0.7
	r_{max}	5
UAV	Altitude	1000 m
	Antenna	Omni-directional

The overall network layout in the Riverbed Modeler is shown in Figure 7. We applied the urban propagation model provided by the Riverbed Modeler when considering the network connectivity from the building attenuation effect. Initially, 11 radio nodes (10 GVNs and one URN) are deployed within a 1000 m × 1000 m region. Each GVN is randomly placed, and the random way point (RWP) model is applied as the mobility model.

Each GVN generates bidirectional USM, RTS, and COP packets, and each GVN establishes a pair with a random destination for three traffic pairs. The URN performs only the relay role and does not generate traffic except for the routing control message. We conducted the simulation 100 times with a 95% confidence interval.

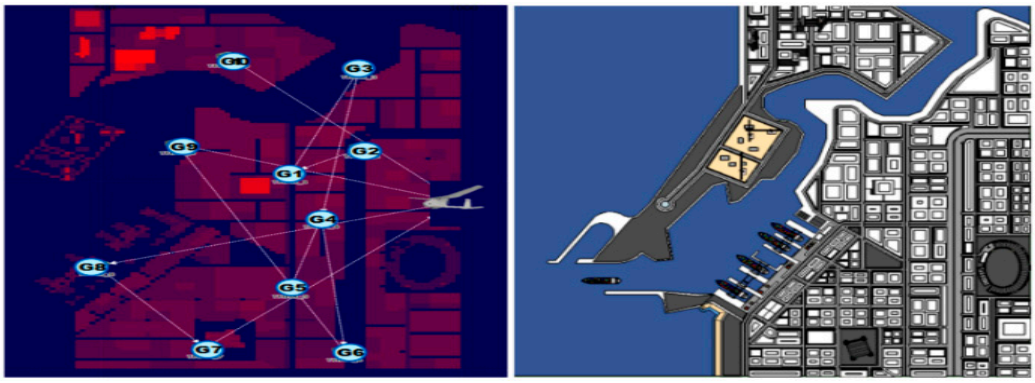


Figure 7. Basic network layout (10 GVNs and 1 URN) in the Riverbed Modeler.

5.2. Performance Analysis

The key element of Q-LBR is URPA, which induces a load balancing between the URN and the ground network. GNs determine the routing according to the URN load and ground network load based on the URPA. Therefore, if the URN grants the maximum allowable traffic through the proper URPA, a positive effect on the overall network performance can be expected because the URN path can provide a higher quality clear-LoS link than the ground path.

Figure 8 shows the results of a comparative experiment when setting the URPA as a fixed value without a learning process and assigning a dynamic value through Q-learning from the perspective of the URN utilization ($Q_{ground_{th}}$= 70, $URCI_{th}$= 80, $GNCI_{th}$= 50, w_1 = 0.7, and w_2= 0.3). URN utilization is a performance index that indicates the average queue length compared to the maximum queue length of the UAV per unit time, and is the same as $URCI_u$, which indicates the queue load of the URN. This metric shows the degree of URN utilization in the network. A lower URN utilization means that $URCI_u(t)$ is low because the load on the URN is idle. By contrast, in the case of the same traffic condition, a higher URN utilization means the UAV load is close to the maximum allowable queue length, and thus the URN is busy. However, if MQL_u is exceeded, it means that a queue drop occurs, and thus it is necessary to set the appropriate $URCI_{th}$.

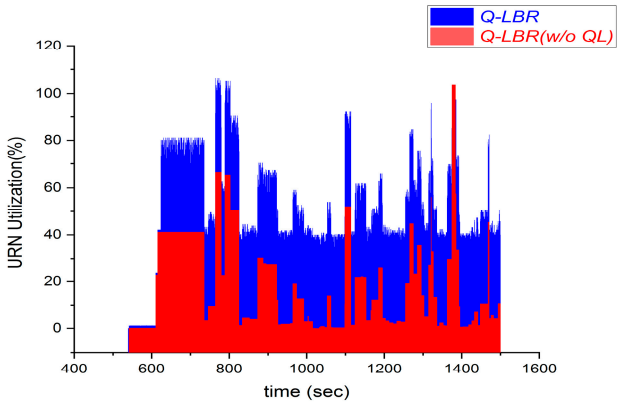


Figure 8. Q-LBR versus Q-LBR without Q-Learning for URN utilization.

In the case of Q-LBR (w/o QL), a fixed URPA policy is applied, and thus there is no coordination according to the ground network load and URN load conditions. Therefore, the overall URN utilization is relatively low (1%–40%). In the case of Q-LBR with Q-learning, the result shows that the URN utilization by dynamic URPA is improved by Q-learning. Therefore, the overall URN utilization is relatively high (40%–80%). If $URCI_{th}$ and $GNCI_{th}$ are increased, a higher URN utilization can be expected in Q-LBR with Q-learning. However, as the URN utilization increases, the possibility of a packet loss owing to an overload increases proportionally, and thus it is necessary to set an appropriate

level (70%–80%). As a result, this experimental result shows that Q-LBR with Q-learning has a significant effect on the dynamic URPA

Figure 9 shows the results of the comparative experiment according to the RCF of Q-LBR in the same environment as the above experiment. The purpose of the experiment was to find out how RCF affects the convergence speed through cumulated reward value (CRV). As a result, it was confirmed that there was a difference in the number of episodes required to reach the maximum reward value (r_{max} = 5) depending on whether or not RCF or the reward value. Q-LBR (w/o RCF, PRF=+1, NRF=−1) approached r_{max} most quickly in the first 10 to 70 episodes, but the results were not converged even after 200 episodes. Q-LBR (w/o RCF, PRF = +0.3, NRF = −0.3) showed convergence after about 160 episodes. This result shows that the probability r_max of is high when the fluctuation of the reward value is small, but the probability of increasing the number of required episodes is high. On the other hand, since Q-LBR (with RCF) adjusted the reward value adaptively in consideration of the ground network load and URN load, it showed a rapid increase in the beginning and converged in a gentle curve. Finally, it converged to 110 episodes, which decreased by about 32% compared to Q-LBR (w/o RCF, PRF = +0.3, NRF = −0.3).

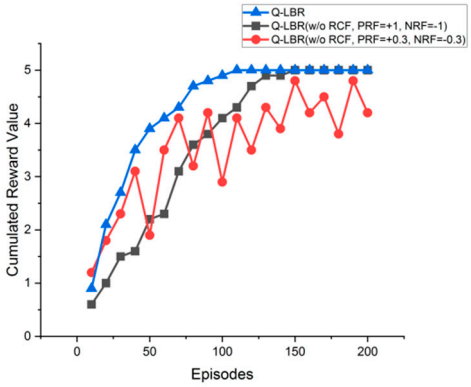


Figure 9. Q-LBR versus Q-LBR without Reward Control Function (RCF) for Cumulated Reward Value.

From Figure 10 and Table 1, we can see that as the node speed increases the packet loss rate of the Q-LBR is lower than that of U2RV. Q-LBR also performs better in terms of network utilization and latency. As the speed of the GVN increases, the probability of the topology changing increases and retransmission by routing control messages and route disconnection increases. U2RV is a multi-criteria routing protocol based on segment density and distance. This protocol only considers the possibility of increasing the traffic through the segment density and does not consider the actual user traffic that may occur in each GVN. In particular, it can be seen that an increase in retransmissions due to a topology change under the same URN coverage may degrade the total network performance.

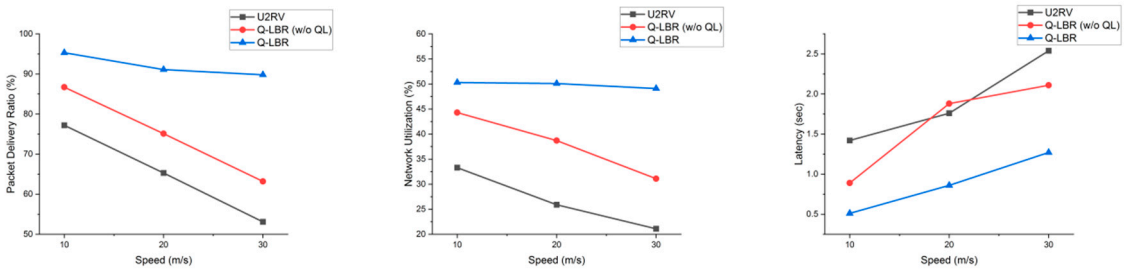


Figure 10. Performance comparison for ground node speed: (**a**) total PDR, (**b**) total network utilization, and (**c**) total latency.

Q-LBR (w/o QL) is the result of setting a fixed URPA value ($URPA_{\_upper}$ = 60, $URPA_{\_lower}$ = 10), except for the Q-learning process. Compared to U2RV, although there is an improvement in performance owing to traffic distribution, a problem occurs in that it is not possible to increase the utilization of the URN by adapting to changes in the network environment. The resulting latency is compared with that of U2RV (20 m/s) in Figure 10c. Based on this result, it can be seen that the fixed URPA may not be properly adapted to the network environment under certain situations.

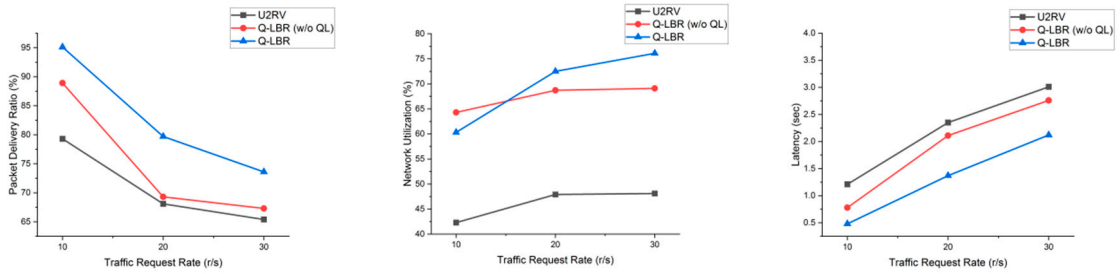
By contrast, Q-LBR shows that it can cope with topology changes caused by network mobility through Q-learning. Q-LBR enables the URPA value to be adaptive to the network situation based on the learning process through RCF. As a result, the changing trend of the graph as the speed increases shows a rather gentle curve compared to the other results. In Table 3, Q-LBR shows a lower COP performance than that of U2RV. This is because COP packets are dropped under congestion or routed only through the ground path by the dynamic URPA. From a system perspective, because COP is a service that is less sensitive to delay and loss, it is reasonable to prioritize USM and RTS. Based on a moving speed of 30 m/s and total traffic flows, Q-LBR shows a PDR of approximately 89.8%, network utilization of 49.1% and latency of 1.27 s.

Table 3. Simulation results for varying the speed of the nodes (all traffic = 10 r/s).

Protocol	Speed (m/s)	Traffic Type	Packet Delivery Ratio (%)	Network Utilization (%)	Latency (s)
U2RV	10	USM	78.3		1.3
		RTS	77.1	33.6	1.5
		COP	79.7		1.6
	20	USM	65.4		1.5
		RTS	69.1	27.3	1.8
		COP	67.3		1.9
	30	USM	66.7		2.1
		RTS	63.5	24.3	2.7
		COP	61.1		2.8
Q-LBR (w/o QL)	10	USM	83.1		0.8
		RTS	82.4	44.8	1.1
		COP	50.5		1.2
	20	USM	81.7		1.4
		RTS	78.5	42.1	1.9
		COP	52.1		2.1
	30	USM	72.1		1.9
		RTS	68.5	28.6	2.1
		COP	58.7		2.2
Q-LBR	10	USM	93.3		0.5
		RTS	91.1	50.3	0.8
		COP	67.5		1.8
	20	USM	92.6		0.6
		RTS	90.8	50.1	0.9
		COP	62.1		2.3
	30	USM	92.5		0.8.
		RTS	89.7	49.8	0.9
		COP	61.8		2.8

Figure 11 and Table 4 show the performance results in terms of the traffic request rate (requests/s), which were similar to those obtained in a previous simulation. However, in the case of a large number of traffic requests exceeding the network capacity, the load balancing efficiency is reduced owing to the multihop resource occupancy of low-priority traffic. This result shows that the dropping of packets in the first hop of the bottleneck link through the URN is more advantageous than dropping through a multihop ground relay. This problem can be solved using the QoS technique (e.g., shaping or policing) to limit the amount of traffic output transmitted with a low priority. Based on the traffic request rate of 30 r/s and the total traffic flows, Q-LBR shows a PDR of approximately 73.6%, a network

utilization of 76.1% and a latency of 2.12 s. As the amount of traffic increases, the overall performance is lowered compared to the previous experiment, but still shows a stable performance based on dynamic load balancing.



(**a**) (**b**) (**c**)

Figure 11. Performance comparison for traffic request rate: (**a**) total PDR, (**b**) total network utilization and (**c**) total latency.

Table 4. Simulation results for varying the traffic request rate (speed = 0 s).

Protocol	Traffic (r/s)	Traffic Type	Packet Delivery Ratio (%)	Network Utilization (%)	Latency (s)
U2RV	10	USM	79.2		1.24
		RTS	76.3	42.8	1.31
		COP	78.1		1.35
	20	USM	68.5		2.11
		RTS	67.6	47.7	2.35
		COP	68.1		2.42
	30	USM	64.3		2.72
		RTS	66.5	47.9	2.77
		COP	66.8		2.83
Q-LBR (w/o QL)	10	USM	88.9		0.81
		RTS	86.5	64.9	0.92
		COP	75.4		0.98
	20	USM	76.3		1.98
		RTS	72.7	67.4	2.37
		COP	64.5		2.61
	30	USM	69.2		2.57
		RTS	65.2	67.3	2.81
		COP	60.7		2.99
Q-LBR	10	USM	96.7		0.44
		RTS	94.2	60.8	0.56
		COP	64.3		0.99
	20	USM	87.4		1.18
		RTS	83.5	72.5	1.22
		COP	62.2		2.76
	30	USM	77.5		1.39
		RTS	74.1	75.9	1.98
		COP	61.9		2.95

6. Discussions

In this chapter, we discuss the feasibility in a real-world scenario of this study. Q-learning faces a problem of memory and high computation requirements if the combination of states and actions are too large. In this paper, network simulation was performed based on 10 GVNs and 1 URN. Computational operations related to Q-learning were performed entirely by URN and there was no problem in running the simulation. However, if the size of the network increases and the number of Q-learning actions increases, the size of the Q table becomes extremely large. In this case it may not be possible to apply the Q-learning algorithm because of the URN's computational power. In particular,

the communication hardware mounted on URN is an embedded system and there are limitations on memory and power. As a solution to this, deep reinforcement learning (DRL), which combines deep learning and reinforcement learning, is considered to be an effective alternative. For example, multistep learning- Deep Q-learning Network (DQN) [24] proposed the concept of using multilayered compensation after a one-step bootstrap when calculating the target Q value. If Q-learning is performed in advance by using the reward information after an n-step bootstrap, it is expected that the amount of computation required for learning can be greatly reduced.

7. Conclusions

In this paper, we proposed a new UAV-assisted routing protocol, called the Q-LBR, that uses a Q-learning algorithm to handle UAV relay traffic. The proposed protocol uses an URPA mechanism when considering the traffic priority and the existence of a route independently from ground network routing. We also proposed an RCF for rapid learning feedback of the reward values in consideration of a dynamic network environment. Q-LBR adjusts the reward value according to the URN load and ground network congestion. Performance evaluation using the Riverbed Modeler showed that Q-LBR achieved a significantly better network throughput and latency compared to that of existing algorithms. As a continuation of this work we plan to continue research on implementation of actual equipment and additional algorithms linked to DRL.

Author Contributions: Conceptualization, B.-S.R. and M.-H.H.; methodology, B.-S.R.; software, M.-H.H.; validation, K.-I.K., B.-S.R., and J.-H.H.; formal analysis, M.-H.H.; investigation, B.-S.R.; resources, J.-H.H.; data curation, B.-S.R.; writing—original draft preparation, B.-S.R.; writing—review and editing, B.-S.R.; visualization, M.-H.H.; supervision, K.-I.K.; project administration, J.-H.H.; funding acquisition, J.-H.H. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Agency for Defense Development.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kashyap, A.; Ghose, D.; Menon, P.P.; Sujit, P.; Das, K. UAV aided dynamic routing of resources in a flood scenario. In Proceedings of the 2019 International Conference on Unmanned Aircraft Systems (ICUAS), Atlanta, GA, USA, 11–14 June 2019; pp. 328–335.
2. Zeng, F.; Zhang, R.; Cheng, X.; Yang, L. UAV-assisted data dissemination scheduling in VANETs. In Proceedings of the 2018 IEEE International Conference on Communications (ICC), Kansas City, MO, USA, 20–24 May 2018; pp. 1–6.
3. Cheng, C.-M.; Hsiao, P.-H.; Kung, H.T.; Vlah, D. Maximizing throughput of UAV-relaying networks with the load-carry-and-deliver paradigm. In Proceedings of the 2007 IEEE Wireless Communications and Networking Conference, Kowloon, China, 11–15 March 2007; pp. 4417–4424. [CrossRef]
4. Oubbati, O.S.; Lakas, A.; Zhou, F.; Güneş, M.; Lagraa, N.; Yagoubi, M.B. Intelligent UAV-assisted routing protocol for urban VANETs. *Comput. Commun.* **2017**, *107*, 93–111. [CrossRef]
5. Oubbati, O.S.; Chaib, N.; Lakas, A.; Bitam, S.; Lorenz, P. U2RV: UAV-assisted reactive routing protocol for VANETs. *Int. J. Commun. Syst.* **2019**, *33*, e4104. [CrossRef]
6. Yang, L.; Yao, H.; Wang, J.; Jiang, C.; Benslimane, A.; Liu, Y. Multi-UAV Enabled Load-Balance Mobile Edge Computing for IoT Networks. *IEEE Internet Things J.* **2020**, *7*, 1. [CrossRef]
7. Guo, Y.; Li, X.; Yousefi'Zadeh, H.; Jafarkhani, H. UAV-aided cross-layer routing for MANETs. In Proceedings of the 2012 IEEE Wireless Communications and Networking Conference (WCNC), Paris, France, 1–4 April 2012; pp. 2928–2933. [CrossRef]
8. Gao, Y.; Zhang, Z.; Zhao, D.; Zhang, Y.; Luo, T. A hierarchical routing scheme with load balancing in software defined vehicular ad hoc networks. *IEEE Access.* **2018**, *6*, 73774–73785. [CrossRef]

9. Yao, H.; Yuan, X.; Zhang, P.; Wang, J.; Jiang, C.; Guizani, M. A machine learning approach of load balance routing to support next-generation wireless networks. In Proceedings of the 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC), Tangier, Morocco, 24–28 June 2019; pp. 1317–1322.
10. Jang, B.; Kim, M.; Harerimana, G.; Kim, J.W. Q-Learning algorithms: A comprehensive classification and applications. *IEEE Access* **2019**, *7*, 133653–133667. [CrossRef]
11. Simon, P. *Too Big to Ignore: The Business Case for Big Data*; Wiley: Hoboken, NJ, USA, 2013; Volume 72.
12. Mammeri, Z. Reinforcement learning based routing in networks: Review and classification of approaches. *IEEE Access* **2019**, *7*, 55916–55950. [CrossRef]
13. Littman, M.L. Reinforcement learning improves behavior from evaluative feedback. *Nature* **2015**, *521*, 445–451. [CrossRef] [PubMed]
14. Jung, W.-S.; Yim, J.; Ko, Y.-B. QGeo: Q-Learning-based geographic ad hoc routing protocol for unmanned robotic networks. *IEEE Commun. Lett.* **2017**, *21*, 2258–2261. [CrossRef]
15. Jafarzadeh, S.Z.; Yaghmaee, M.H. Design of energy-aware QoS routing protocol in wireless sensor networks using reinforcement learning. In Proceedings of the 2014 IEEE 27th Canadian Conference on Electrical and Computer Engineering (CCECE), Toronto, ON, Canada, 4–7 May 2014; pp. 1–5.
16. Dong, S.; Agrawal, P.; Sivalingam, K.M. Reinforcement learning based geographic routing protocol for uwb wireless sensor network. In Proceedings of the IEEE GLOBECOM 2007—IEEE Global Telecommunications Conference, Washington, DC, USA, 26–30 November 2007; pp. 652–656. [CrossRef]
17. Yang, Q.; Jang, S.-J.; Yoo, S.-J. Q-learning-based fuzzy logic for multi-objective routing algorithm in flying ad hoc networks. *Wirel. Pers. Commun.* **2020**, *113*, 115–138. [CrossRef]
18. Al-Hourani, A.; Kandeepan, S.; Jamalipour, A. Modeling air-to-ground path loss for low altitude platforms in urban environments. In Proceedings of the 2014 IEEE Global Communications Conference, Austin, TX, USA, 8–12 December 2014; pp. 2898–2904. [CrossRef]
19. Mozaffari, M.; Saad, W.; Bennis, M.; Debbah, M. Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage. *IEEE Commun. Lett.* **2016**, *20*, 1647–1650. [CrossRef]
20. Lijun, D.; GANG, W.; JINGWEI, F.; Yizhong, Z.; YIFU, Y. Joint Resource Allocation and Trajectory Control for UAV-Enabled Vehicular Communications. *IEEE Access.* **2019**, *7*, 132806–132815. [CrossRef]
21. Jiang, D.; Delgrossi, L. IEEE 802.11p: Towards an international standard for wireless access in vehicular environments. In Proceedings of the VTC Spring 2008—IEEE Vehicular Technology Conference, Singapore, 11–14 May 2008; pp. 2036–2040. [CrossRef]
22. Ahmed, A.; Sidi-Mohammed, S.; Samira, M.; Hichem, S.; Mohamed-Ayoub, M. *Efficient Data Processing In Software-Defined Uav-Assisted Vehicular Networks: A Sequential Game Approach*; Springer Wireless Personal Comm.: Berlin/Heidelberg, Germany, 2018; Volume 101, pp. 2255–2286.
23. Jobaer, S.; Zhang, Y.; Hussain, M.A.I.; Ahmed, F. UAV-assisted hybrid scheme for urban road safety based on VANETs. *Electronics* **2020**, *9*, 1499. [CrossRef]
24. Yuan, Y.; Yu, Z.L.; Gu, Z.; Yeboah, Y.; Wei, W.; Deng, X.; Li, J.; Li, Y. A novel multi-step Q-learning method to improve data efficiency for deep reinforcement learning. *Knowl. Based Syst.* **2019**, *175*, 107–117. [CrossRef]