PAPER

# Distributed Reinforcement Learning Approach for Vehicular Ad Hoc Networks

**Celimuge WU**[†a)], ***Student Member***, **Kazuya KUMEKAWA**[†], ***and*** **Toshihiko KATO**[†], ***Members***

**SUMMARY** In Vehicular Ad hoc Networks (VANETs), general purpose ad hoc routing protocols such as AODV cannot work efficiently due to the frequent changes in network topology caused by vehicle movement. This paper proposes a VANET routing protocol QLAODV (Q-Learning AODV) which suits unicast applications in high mobility scenarios. QLAODV is a distributed reinforcement learning routing protocol, which uses a Q-Learning algorithm to infer network state information and uses unicast control packets to check the path availability in a real time manner in order to allow Q-Learning to work efficiently in a highly dynamic network environment. QLAODV is favored by its dynamic route change mechanism, which makes it capable of reacting quickly to network topology changes. We present an analysis of the performance of QLAODV by simulation using different mobility models. The simulation results show that QLAODV can efficiently handle unicast applications in VANETs.

***key words:*** *vehicular ad hoc networks, routing protocol, Q-Learning, AODV*

## 1. Introduction

A Vehicular Ad hoc Network (VANET) is a form of mobile ad hoc network providing communications between vehicles in close proximity, and between vehicles and nearby fixed roadside equipment. The opportunities for VANETs are growing rapidly. Chaabouni et al. [1] give an overview of some inter-vehicular applications and their main characteristics. According to [1], inter-vehicular applications can be classified into four main application groups: traffic safety, floating car data, Internet access and group communication. Many broadcast protocols and multicast protocols have been proposed for traffic safety and floating car data applications [2]. However, in this paper we mainly consider the unicast routing protocols that can be used in inter-vehicular applications. In VANETs, vehicles can utilize the Internet with the help of other vehicles even though they can not communicate with an access point directly. Vehicles also can use multi-hop communication to share information with other vehicles.

The main distinctive features of vehicular ad hoc networks are high mobility and frequent link changes. Thus, the routing problem of finding reliable paths from a traffic source to a traffic destination through a series of intermediate forwarding nodes is particularly challenging. It is therefore crucial to design an efficient routing protocol for VANETs. Li and Wang [3] have discussed the research challenge of routing in VANETs and surveyed recent routing protocols. Generally, VANET routing protocols in inter-vehicular unicast applications can be classified into two different approaches: position-based and topology-based routing. In position-based routing, the routing decision at each node is based on the destination's position and the position of the forwarding node's neighbors [4], [5]. Maintaining position information needs additional control packets which leads to bandwidth wastage. Therefore, the performance of position-based routing is limited by high control overheads. In contrast, topology-based routing protocols use the information about the links that exist in the network to perform packet forwarding. Although there are several topology-based routing protocols which have been designed for VANET, they all have their limitations.

This paper proposes a design of a general inter-vehicular unicast routing protocol whose purpose is to react quickly to node mobility and topology changes. We propose an enhanced reinforcement learning [6] routing protocol QLAODV (Q-Learning AODV). QLAODV uses Q-Learning [7], a recent form of reinforcement learning algorithm, to infer network link state information and to dynamically change routes according to the information learned. By taking advantage of reactive route discovery, periodic partial exchange of link state data and preemptive route changing, QLAODV meets the requirements of inter-vehicular unicast applications very well.

The remainder of the paper is organized as follows. In Sect. 2, we give a brief description of related work and elaborate our contribution. In Sect. 3, we give a detailed description of the proposed routing protocol QLAODV. Next, we present simulation results and discussions in Sect. 4. Finally, we present our conclusions in Sect. 5.

## 2. Related Work and the Contribution of This Study

### 2.1 Routing Protocols in VANETs

As mentioned above, VANET unicast routing protocols are classified into position-based routing and topology-based routing. In this paper we suggest using a topology-based routing protocol in VANET unicast applications. There are several reasons why we prefer a topology-based approach. Firstly, a topology-based approach does not depend on particular instruments such as GPS positioning devices, which are not affordable for use in every vehicle. Secondly,

Manuscript received July 6, 2009.
Manuscript revised January 9, 2010.

[†]The authors are with the Department of Information Network Science, Graduate School of Information Systems, University of Electro-Communications, Chofu-shi, 182-8585 Japan.
a) E-mail: clmg@net.is.uec.ac.jp

DOI: 10.1587/transcom.E93.B.1431

position-based routing protocols have not been able to produce fully satisfactory results. Many performance comparisons between position-based and topology-based routing protocols assume that nodes can determine the location of their neighbors and of the destination (e.g., GPSR [8], MURU[9]). In highly dynamic networks, to get precise position information about other nodes, large numbers of signaling packets are needed. This incurs a greater traffic overhead and leads to performance deterioration.

Topology based routing in VANETs has been studied recently and many protocols have been proposed [10]–[14]. A prediction-based routing (PBR) protocol is introduced by Namboodiri and Gao [10]. However, it is not applicable to street scenarios. Taleb et al. [11] introduce a scheme which groups vehicles according to their direction of movement. However, in the case of winding roads (e.g., mountainous areas), the approach of grouping vehicle on the basis of their velocity vector is inadequate. Yang et al. [12] present the connectivity aware routing protocol. The disadvantage of this method is the increase in packet size and the impact on network bandwidth. Ducourthial et al. [13] present an approach for routing in highly dynamic networks, relying on condition-based communication. The main drawback of this approach is its application dependency. Lu et al. [14] give a thorough discussion on the feasibility of enhancing the network performance by the use of buses, street lamps and traffic lights as the bridge nodes in a city area. However, in this paper, we aim to propose a general VANET routing protocol that does not rely on such bridge nodes.

It appears to be more promising to modify an existing routing protocol than to design a new protocol from scratch. AODV [15] is known as a good performer in MANET routing protocols. There have been several research attempts which try to extend use of the AODV protocols to VANETs [16]–[18]. Menouar et al. [16] improve the AODV routing process by selecting the most stable route with respect to the movement of the vehicles. However, this study is only interested in the route discovery process, and so can not adapt quickly to frequent topology changes. Wang et al. [17] introduce a Two-Phase routing protocol (TOPO) that incorporates map information in routing. However, this does not work well in small scale scenarios. Moreover, TOPO is not suitable for high data rate traffic because it faces the problem of wireless channel congestion in overlay. Abedi et al. [18] propose the DAODV protocol that uses two parameters, direction of movement and vehicle position, to select the next hop during the route discovery phase. However, this does not work in street scenarios. Additionally, other studies [11], [12], [16]–[18] assume that every node knows its own position while Wang [17] assume that map information is also available.

### 2.2 Route Errors and Link Breakage Processing

In AODV, when a link break occurs in an active route, the node upstream of that break may try to perform a local repair or send back a route error (RERR) packet to the source node, depending on whether or not local repair is enabled. If local repair is disabled, all the packets that are transmitted between the instant of link failure and the reception of RERR at the source are dropped. If local repair is used, the upstream intermediate node tries to establish a new route segment from itself to the destination. However, the local repair mechanism has some limitations. First, the condition for invoking local repair is that the destination should be no farther than a preset number of hops away from the broken link. Second, the local repair mechanism introduces route non-optimality, and suffers from frequent link breaks and heavy control overheads in networks with high node mobility.

A scheme, which improves the data delivery fraction of AODV (AODV-HPDF) by utilizing local repair at the upstream intermediate node without the hop-distance condition, has been presented by Liang and Wang [19]. In AODV-HPDF, the node that detected the link break will send an RERR packet to the source node. When the source node has received the RREP packet, it will initiate a route-rediscovery process only if the data transmission still necessary. Also, the node that detected the link break will be treated as a new source node and a route discovery process will be initiated on that node with a limited time-to-live (TTL) RREQ packet and a limited timeout. Once the new temporary primary route has been built successfully by the new source node within the timeout of the RREQ packet, the buffered data packets will be sent to the destination node through it. While providing slightly better performance, AODV-HPDF suffers from a high control overhead in high mobility scenarios.

A novel technique called Neighborhood Route Diffusion (NRD) has been proposed by Quwaider et al. [20]. The key idea is to perform the local diffusion of selective route information to neighbor nodes, in order to create a temporary envelope of emergency route information to a destination around all nodes that are actively forwarding packets to that specific destination. When a link on a route fails due to mobility, the upstream intermediate node on the failed link can forward packets to one of its neighbors, which has already been provided with route information for the corresponding destination. This can salvage packets without having to rely on slow and control-heavy end-to-end and local repair mechanisms. However, the advantage of NRD decreases with an increasing number of destinations because in that case the likelihood of finding routing information for a destination will be lower.

### 2.3 Reinforcement Learning Approaches in Routing Protocols

In recent years, reinforcement learning [6] has been attracting increasing interest in the machine learning and artificial intelligence communities. Boyan and Littman [21] describe the Q-routing algorithm for packet routing, in which a reinforcement learning module is embedded into each node of a switching network. Since Q-routing is designed for wired networks, it is not suitable to VANETs. Chang et al. [22] use reinforcement learning methods to control both packet routing decisions and node mobility to improve the connectivity of a network. However, it is impossible to control node movement in VANETs. Dowling et al. [23] have proposed collaborative reinforcement learning (CRL), which enables groups of reinforcement learning agents to solve system optimization problems online in dynamic, decentralized networks. They evaluate an implementation of CRL in a routing protocol for MANETs, which is called SAMPLE. However, CRL has the problem of convergence to suboptimal solutions. What is more important is that SAMPLE does not consider link breakage due to node mobility which is the main feature of VANETs. Although SAMPLE performs well in high packet error rate scenarios, it has worse packet delivery ratios than AODV in cases where the packet loss due to radio interference is low. In SAMPLE, routing information is advertised in the network by attaching it to data packets. As a result, it increases the data packet size and so introduces a large overhead in high data rate applications.

Based on the original AODV, we present an enhanced routing protocol called QLAODV that uses a Q-Learning algorithm [7] to achieve whole network link status information from local communication and to change routes preemptively using the information so learned. In order to make Q-Learning work efficiently in highly dynamic networks, we propose a route change request/reply mechanism to check the usability of a newly learned route. Through exhaustive simulation, we have confirmed that QLAODV is able to discover better routes in a dynamically changing network without having to know the network topology and traffic patterns in advance, and therefore can adjust quickly to topology changes. To the best of our knowledge, our proposal which uses a reinforcement learning algorithm to optimize VANET routing protocol is being studied the first time.

## 3. QLAODV Protocol Design

In this section, we present a detailed description of our proposed protocol QLAODV. QLAODV is an enhanced topology-based routing protocol. When a source node needs to communicate with a destination node, it checks its routing table for a route. If none exists, QLAODV uses the normal route discovery approach of AODV to create a route to the destination. To avoid AODV's need for frequent route discovery in highly dynamic networks, we use a dynamic route change mechanism to switch routes preemptively and therefore reduce the number of route request broadcasts. In order to discover a better route, we use Q-Learning, a recent form of reinforcement learning algorithm, to infer network link state information in a distributed manner. Every network node acts as a learning agent and gathers network link state information while interacting with its local environment. We also propose a mechanism to check the availability of a new route. The mechanism supplements the Q-Learning algorithm in order to work efficiently in highly dynamic networks. In order to meet the requirements of inter-vehicular applications, we consider the hop count, stability and bandwidth efficiency in route selection.

### 3.1 Reinforcement Learning Model for VANET Routing

It is difficult to use a simple rule to determine the packet forwarding policy because of frequent link changes in VANETs. Moreover, the frequent topology changes also make it necessary to change the forwarding policy concurrently. Fortunately, the use of reinforcement learning can handle these problems. Reinforcement learning algorithms attempt to find a policy that maps states of a system to the actions that the agent ought to take in the event of those states occurring. In reinforcement learning, the correct input/output pairs are never presented and the evaluation of the system is often concurrent with learning.

Reinforcement learning is the problem faced by an agent who must learn behavior through trial-and-error interactions with a dynamic environment. Formally, the reinforcement learning model consists of: ($a$) a discrete set of environment states, $S$; ($b$) a discrete set of agent actions, $A$; and ($c$) a set of scalar reinforcement rewards, $R$.

In this work, we model the network routing problem in VANETs as follows. The entire vehicular ad hoc network is the environment. Its components include the mobile nodes, the links between the nodes and packets. Each packet $P(o, d)$, indexed by its originator node $o$ and destination node $d$ is an agent. Each node in the network is considered a state of the agent. The set of all nodes in the network is the state space. A node selects the next hop that it should forward a packet to (or delivers it to the upper layer if the current node is the destination node). Hence the possible set of actions allowed at the node is nothing but the set of neighbors. The state transitions are equivalent to a packet being delivered from one node to its neighbor.

Since it is impossible to have a global view on network state transitions, we distribute the reinforcement learning task to each node. Nodes exchange their knowledge through hello messages. Each node only needs to select its best next hop. Upon selecting the next hop, the node should immediately receive back the next hop node's estimate. However, considering the control overhead and implementation complexity, we use periodic hello messages to help nodes to revise their estimates. In QLAODV, the agent might receive a negative reward if the route change attempt fails (this will be explained in 3.6).

### 3.2 Distributed Q-Learning in QLAODV

For VANETs, as a packet is routed, there is no way to determine the reward until the packet reaches the destination node. Hence using the model-based approach is not possible. Therefore, we use Q-Learning [7], which is able to compare the expected utility of the available actions without requiring a model of the environment.

Q-Learning is a recent form of reinforcement learning algorithm that does not need a model of its environment and works by estimating the values of state-action pairs. The Q-value $Q(s, a)$ ($s \in S, a \in A$) in Q-learning is an estimate of the value of future rewards if the agent takes a particular action $a$ when in a particular state $s$. By exploring the environment, the agents build a table of Q-values for each environment state and each possible action. Except when making an exploratory move, the agents select the action with the highest Q-value. The learning rate and the discount factor are important parameters of the Q-learning algorithm. The learning rate parameter limits how quickly learning can occur. It governs how quickly the Q-values can change with each state/action change. The discount factor controls the value placed on future rewards. If the value is low, immediate rewards are optimized, while higher values of the discount factor cause the learning algorithm to count future rewards more strongly.

The Q-Learning algorithm that is used in QLAODV is defined as follows. Every node maintains a Q-Table which consists of Q-values $Q(d, x)$ whose values range from 0 to 1, where $d$ is the destination node and $x$ is the next hop to the destination. We use a dynamic Q-Table, such that the size of the Q-Table of a node is determined by the number of destination nodes and neighbor nodes. The Q-Table and learning tasks are distributed among the different nodes (states). In QLAODV, exploration can be achieved by updating the Q-values when the agent receives a hello message. Therefore, when choosing a next hop, we let the agent act greedily, taking, in each situation, the action with the highest Q-value. If a packet is able reach its destination node through the action $x$, the reward $R$ will be 1, and otherwise $R$ will be 0. More specifically, when a node receives a hello from the destination node, the reward $R$ will 1 and otherwise $R$ will be 0.

The discount factor is an important parameter of the Q-learning algorithm. We use a variable discount factor, which is determined by the hop count, link stability and available bandwidth of nodes on the route. The information will be discounted when it passes through the node and will also be discounted according to link stability and bandwidth usage. In this way, we ensure that the route we select is the shorter, more stable route with enough bandwidth. We estimate the local used bandwidth $BW$ as

$$BW(\text{bps}) = \frac{n \times S_B \times 8}{T}, \tag{1}$$

as defined by Renesse et al. [24]. We assume all nodes have the same maximum bandwidth and therefore we can get the Available Bandwidth by subtracting the local used bandwidth from the Maximum Bandwidth. In Eq. (1), $n$ is the number of packet sent and received by a node. $S_B$ is the size of a packet in bytes while $T$ is the time period. We set $T$ to 0.5 s in our QLAODV implementation.

### 3.3 Maintenance of the Q-Table

In QLAODV, every node uses hello messages to exchange link information with its neighbors. This link information includes a part of the Q-Table (MaxQValues), the mobility factor of the node and the bandwidth factor of the node. In this paper, we define $Q_s(d, x)$ as the Q-Metric of node $s$ bound to destination node $d$ through neighbor $x$.

In QLAODV, when the hello timer expires, every node first calculates an array (MaxQValues) which contains maximum Q-Metrics for each destination node in the network.

Every node $x$ then calculates a mobility factor $MF_x$ as

$$MF_x = \begin{cases} \sqrt{\frac{|N_x \cap N_x^p|}{|N_x \cup N_x^p|}}, & \text{if } N_x \cup N_x^p \neq \phi \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where $N_x$ is the current neighbor set of node $x$ and $N_x^p$ denotes the neighbor set of node $x$ at the time that the previous hello was sent. Every node needs to maintain a $N_x^p$. When the hello timer expires a node uses this value and the current neighbor set to calculate $MF_x$. The $MF_x$ will reflect a higher value for a relatively stable node. In case of a static network, $MF_x$ will be 1 for every node.

Every node $x$ also needs to calculate a bandwidth factor $BF_x$ as

$$BF_x = \frac{\text{Available Bandwidth of } x}{\text{Maximum Bandwidth of } x.} \tag{3}$$

Every node then attaches the MaxQValues, $MF_x$ and $BF_x$ to the hello message.

We assume that at the start of communication, agents know nothing about the rest of the network. This means that all elements of Q-Table (Q-values) are initialized to 0. $Q_s(d, x)$ is the value that node $s$ estimates as the practicability of delivery of a packet bound for node $d$ by way of neighbor node $x$. This estimation represents the whole network performance because it considers multiple metrics of hop count, stability and bandwidth division. Upon receiving a hello packet from the neighbor $x$, a node first calculates a discount factor $\gamma_x$ as

$$\gamma_x = \gamma \times \sqrt{MF_x \times BF_x} \tag{4}$$

where $\gamma$ is a predefined value. $\gamma_x$ should satisfy $0 < \gamma_x < 1$ to consider the hop count. Because the mobility factor ($MF_x$) and bandwidth factor ($BF_x$) are considered in $\gamma_x$'s calculation, we set $\gamma$ to the relatively large value of 0.9. The node $s$ then revises its estimate as

$$\begin{aligned} Q_s(d, x) \leftarrow\ & (1 - \alpha) Q_s(d, x) + \\ & \alpha \left\{ R + \gamma_x \max\nolimits_{y \in N_x} Q_x(d, y) \right\} \end{aligned} \tag{5}$$

where $N_x$ denotes the set of neighbors of node $x$ and $R$ denotes the reward. $R$ is defined as

$$R = \begin{cases} 1, & \text{if } s \in N_d \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where $N_d$ is the set of neighbors of $d$. This means that if a node receives a hello from the destination, the reward will be 1 and otherwise 0. In Eq. (5), $\max_{y \in N_x} Q_x(d, y)$, actually an element of MaxQValues, is calculated by the hello sender node and sent with its hello message. In this way, a hello sender node does not need to send the whole Q-Table and hence can minimize the hello overhead.

The learning rate parameter $\alpha$ limits how quickly learning can occur. In the proposed protocol, it governs how quickly the Q-values can change with a network topology change. If the learning rate is too low, the learning will not adapt quickly to network dynamics. If the rate is too high, then the algorithm cannot reflect the network movements accurately because agents can receive immediate misleading rewards. We set the learning rate $\alpha$ to 0.8. This is because we know this is the most suitable value for our protocol after a lot of experiments and analysis.

The nodes exchange link state information and update their Q-Table using hello messages. Each node attaches its MaxQValues, $MF_x$ and $BF_x$ to the hello message before sending it. The node that receives the hello message extracts the corresponding values from the hello packet and executes the Q-Learning algorithm to update its Q-Table. The MaxQValues that a node obtains from a received packet is the Q-Metrics of the neighbor who sent it and it indicates the neighbor's knowledge about the network.

### 3.4 Exploitation, Exploration and Convergence

When forwarding data, QLAODV selects for the next hop the node that has maximum Q-value. This is called exploitation. Nevertheless, to make the exploitation lead to the global optimum, an exploration is required to check whether one neighbor is better than another. In QLAODV, each node updates its Q-values upon reception of hello messages from its neighbors. Since hello messages are exchanged periodically, every node is aware of which neighbor is becoming the preferred choice.

Convergence is an important issue in evaluating an algorithm's validity. There is no guarantee that reinforcement learning always leads to convergence. However, Watkins and Dayan [25] prove that Q-Learning converges to the optimum action-values with probability of 1 so long as all actions are repeatedly sampled in all states and action-values are represented discretely. Fortunately, our algorithm satisfies all the conditions for convergence. In the proposed algorithm, a node is equivalent to a state and every node uses hello messages to sample all its neighbors. Obviously, the action-values (Q-values) are represented discretely in QLAODV. Therefore, we can prove our proposed algorithm converges to the optimum action-values.

### 3.5 Routing Metrics in QLAODV

Many Distance vector routing protocols such as AODV try to find the shortest route possible. However, the shortest route is not always the best route. QLAODV uses the Q-Learning algorithm to evaluate a path according to its hop count, stability and available bandwidth. QLAODV gives a shorter path a higher value because the discount factor $\gamma_x$ is smaller than 1. Since QLAODV considers the mobility factor, $MF_x$, in the calculation of discount factor, it can choose the most stable route. Stability is also reflected in the Q-Metric through the value iteration. As shown in Eq. (5), for the first calculation, $Q_s(d, x)$ is zero and this value is discounted by $1 - \alpha$ for every iteration. This means that $Q_s(d, x)$ is expected to become larger with each iteration if other elements do not change. In general, if a link's duration time is long, it is more likely to still be durable in the future which is the case of a vehicle traveling in the same direction. QLAODV can also balance the traffic between nodes because it discounts the reward according to the available bandwidth. In short, QLAODV can achieve short, stable and high-bandwidth routes.

### 3.6 Dynamic Route Change Mechanism to Avoid Link Breakage

It is possible that the route learned from local communication is already out-of-date because of link breakage in a fast moving network. In order to check whether the route is still available or not, we use unicast route change request and route change reply messages. When a route is being used for delivering packets, if a sender node (source node or other forwarder node) finds an alternative path that has a larger Q-Metric than the current route, the sender node will send a unicast packet RCNG-REQ (route change request) to the destination through the neighbor which indicates a better route to the destination. The intermediate nodes will forward the packet according to their Q-Table. Upon receiving the RCNG-REQ packet, the destination node replies with RCNG-REP (route change reply). This means the new path is available if the RCNG-REP reaches the sender node successfully. The sender node then updates its routing table to use the new route. Every forwarder node also updates the corresponding route upon receiving a RCNG-REP. The compositions of the RCNG-REQ packet and the RCNG-REP packet are shown in Table 1. Fig. 1 depicts the dynamic route change approach of QLAODV.

As shown in Fig. 1, we assume that node s uses next hop 1 to deliver data packets bound for destination node d.

**Table 1** Composition of RCNG-REQ packet and RCNG-REP packet.

| RCNG-REQ | RCNG-REP |
|---|---|
| Destination IP Address | Destination IP Address |
| Destination Seq Number | Destination Seq Number |
| Originator IP Address | Originator IP Address |
| Originator Seq Number | Life Time |
| Next Hop | Next Hop |

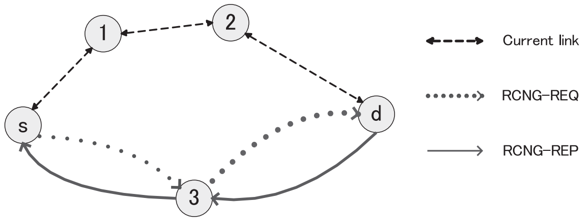


**Fig. 1** Dynamic route change mechanism in QLAODV.

We note that node s could be the source node or another forwarder node. Each node offers connectivity information by broadcasting hello messages, and utilizes the Q-Learning algorithm to update its own Q-Table when it receives hello messages from its neighbors. Node s updates its Q-Table upon receiving a hello packet from node 3. Node s then checks its Q-Table and finds that the new path by way of node 3 to destination node d is better than the current route. In order to check the availability of the new path, node s will set a route change timer and initiate a unicast packet RCNG-REQ to destination node d and send it by using neighbor node 3 as the next hop. Upon receiving the RCNG-REQ packet, node 3 knows the packet is for node d. Node 3 then selects the best next hop according to its Q-Table to forward the RCNG-REQ packet. Every intermediate node also sets a route change timer before forwarding the RCNG-REQ. As soon as the destination node receives the RCNG-REQ packet, it initiates a RCNG-REP to node s and sends it by way of node 3. In the same way, node 3 will forward it to node s. Node s updates its route table if it receives the RCNG-REP packet before the route change timer expires. In this way, without the original route request being broadcast, node s can use the new route to deliver data and thus can reduce the routing overhead compared with other approaches and consequently improve the data delivery ratio. Conversely, if a node (including RCNG-REQ sender node and other forwarder nodes) does receive the route change reply before the route change timer expires, the corresponding Q-value will be reset to 0.

## 4. Simulation Results

We used Network Simulator 2 (ns-2) to conduct simulations using different mobility models. First, we used the Freeway mobility model and the Manhattan mobility model [26] to evaluate the protocols' performance. The Freeway mobility model emulates the motion behavior of mobile nodes on a freeway while the Manhattan mobility model emulates the movement pattern of mobile nodes on smaller side streets. In the freeway model simulation, we use a freeway which has two lanes in each direction. All lanes of the freeway are 2000 m in length. 80 vehicles are randomly distributed on this freeway and the arrival velocity of each vehicle is 5 m/s. For each of the Manhattan model scenarios, we use a map of 80 nodes randomly distributed in a street area of 1000 m × 1000 m. The map consists of 3 horizontal streets and 3 vertical streets and every street has one lane in each direction. The distance between intersections is 300 m. We set the arrival velocity to 5 m/s. Next, we use a Tiger line map file [27] and real street map based model [28] to generate realistic vehicle movement scenarios. We use a 2500 m × 2500 m square area in Midtown Manhattan in New York City as shown in Fig. 2. We choose this area because it is representative of a large number of city areas in the US. In the freeway mobility model and the Manhattan mobility model, the transmission range is 250 m. Nevertheless, in the real street map based mobility model, we use a 500 m transmission range, as is suggested in [28].

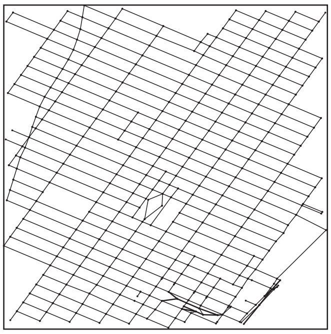


**Fig. 2** Street scenario corresponding to a 2500 m × 2500 m square area in midtown Manhattan.

The QLAODV protocol was compared with AODV and two other extensions of the AODV protocol (AODV-HPDF and NRD). In all simulations, we assume omnidirectional antennas, IEEE 802.11b standard transmission at 11 Mbps and standard 802.11 MAC. We use link layer notification as provided by 802.11 to determine connectivity. The standard CMUPri model for a queue of buffer size 50 was used. We used CBR traffic with a packet size of 512 bytes and UDP when running the simulation. Each simulation lasted 500 s and each case was repeated 50 times to give high confidence in the results. All data presented in this paper are the average value of the 50 simulations.

### 4.1 Effect of Mobility

In the Freeway Model and the Manhattan Model, each vehicle accelerates at a rate of ten percent of the maximum allowable velocity per second, if there are no other vehicles ahead of it, until the maximum allowable velocity is reached. We simulate various values of maximum allowable velocity in the Freeway Model and the Manhattan Model. In the real street map based model, the speed limit for each road was based on the type of road as indicated in the TIGER/Line files [27]. In addition, we present simulation results with various node densities. For all models, we generated 30 pairs of random connections with a 32 kbps transmission rate. Fig. 3 and Fig. 4 show comparisons of the achieved packet delivery ratio for AODV, AODV-HPDF, NRD and QLAODV for the different mobility models. We calculated the packet delivery ratio as the number of data packets received by the application layer of the destination nodes divided by the number of data packets generated by the source nodes.

It can be clearly seen that QLAODV outperforms the other three protocols, irrespective of the mobility model. In the freeway model, we also can see that as the node velocity increases, the advantage of QLAODV becomes more apparent. This can be explained by the fact that in dynamically changing networks, QLAODV can change to better routes adaptively as the network topology change, whereas other protocols wait until existing routes break before constructing new routes. Also, since QLAODV takes the stability of

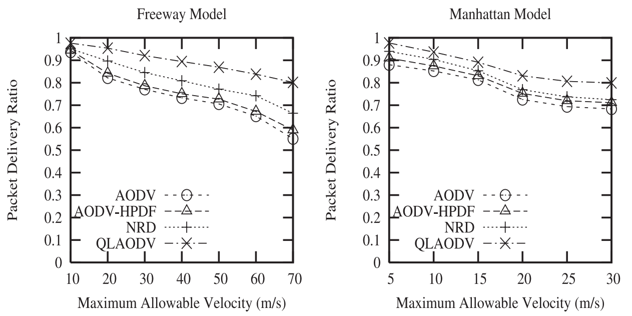


**Fig. 3** Achieved packet delivery ratio for varying velocities in Freeway model (Left) and Manhattan model (Right).

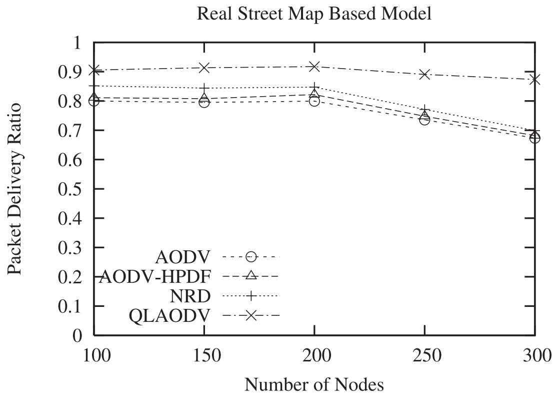


**Fig. 4** Achieved packet delivery ratio for varying number of nodes in real street map based mobility model.

routes into account, it results in a high probability of using vehicles moving in the same direction as the source node to forward packets. However, in AODV, a source node initiates a route request to broadcast the packet and the destination node simply replies with the route which has the minimum hop count. Therefore, a node which is moving in the opposite direction to the source node may be chosen as an intermediate node, and in this case the corresponding route is very vulnerable. Consequently, many data packets may be dropped when link failure occurs. NRD may use oncoming vehicles or vehicles moving in other direction to salvage data packets. This results in a significantly higher frequency of route failures. AODV-HPDF also suffers from the same problem, because AODV-HPDF's local repair always leads to non-optimal paths. In QLAODV, vehicles moving in the same direction as the source node always retain a higher Q-Metric than those moving in other directions. Thus, vehicles can use other vehicles moving in the same direction to forward data. Therefore, QLAODV is more efficient than the other three protocols.

In the Manhattan model, QLAODV clearly outperforms the other three protocols in terms of packet delivery ratio even when the vehicles' moving velocity is very low. This can be explained by the following facts. Even when vehicles' velocity is not very high, the relative speed between vehicles may still be high and this results in frequent topology changes. While the other three protocols can not adapt quickly to network topology changes, QLAODV benefits from its preemptive route change mechanism. Also, since the Q-Learning algorithm takes the hop count into consideration, QLAODV always constructs a shorter route than AODV (as discussed later and shown in Fig. 13 and Fig. 14). This is another factor contributing to QLAODV's advantage. AODV-HPDF and NRD show a decrease in advantage over AODV as a result of increasing mobility. This is because AODV-HPDF and NRD result in longer routes which are easily broken in Manhattan scenarios.

The results for the real street map based mobility model are similar to those for the other mobility models. In AODV-HPDF, upon the occurrence of a link failure, both the upstream node and the source node initiate route discovery. This will become very costly in terms of overheads in high-density networks. This is why AODV-HPDF's advantage decreases with increasing node density. As the number of nodes increases, the flows become more distributed and hence the effectiveness of NRD diminishes. With AODV, when the node density is high, many link failures occur and route request broadcasts consume more bandwidth, leading to a drop in performance. We can also observe that the advantage of QLAODV increases as the number of nodes increases. The reason is that the QLAODV protocol is favored by the increasing number of available paths and it becomes easier to change to a new route before the current one is disconnected.

In AODV, when a link fails, the upstream intermediate node tries to perform a local repair. However, the condition for success of a local repair is that the destination should be no farther than a preset number of hops away from the broken link. If the local repair fails, the buffered packets will be dropped. AODV-HPDF utilizes local repair without the hop-distance condition to improve the packet delivery fraction of AODV. However, while offering faster repairs than the route error based end-to-end mechanisms, local repair introduces route non-optimality, and the new route may fail shortly after the repair. In NRD, when a link on a route fails due to mobility of nodes, the intermediate node on the failed link can forward packets to one of its neighbors which has already had the route information for the corresponding destination diffused to it. However, NRD only works if nodes around the point of failure have routing information to the same destination. In the case where flows are distributed, NRD cannot provide good performance. Moreover, NRD always results in non-optimal paths which diminishes the advantage of NRD.

A comparison of the normalized control overhead is shown in Fig. 5 and Fig. 6. We define the normalized control overhead to be the number of control packets generated divided by the number of data packets that arrive at receivers. In Fig. 5, as the node velocity increases, the control overhead of AODV increases because of route errors and route request broadcasts. We can observe that the normalized control overhead of AODV-HPDF is higher than that of AODV especially at high node velocities. In AODV-HPDF, when a link fails, both the source node and the upstream node ini-

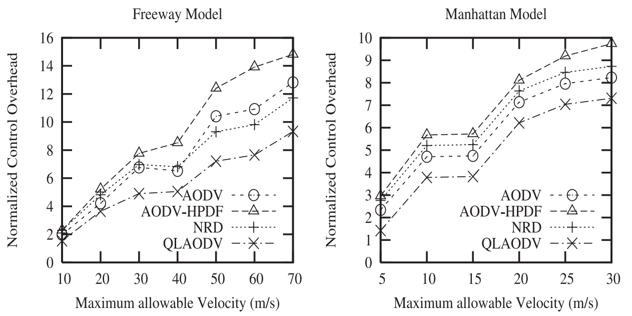


**Fig. 5** Normalized control overhead for varying velocities in freeway model (left) and Manhattan model (right).

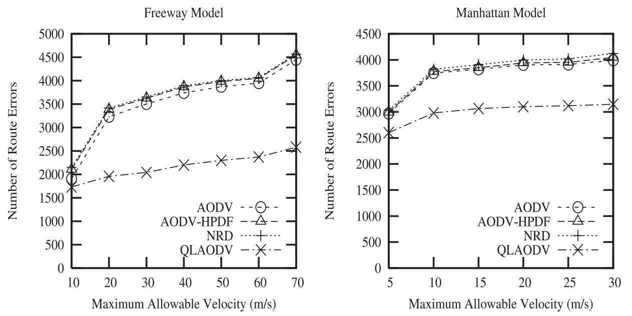


**Fig. 7** Number of route errors for varying velocities in freeway model (left) and Manhattan model (right).

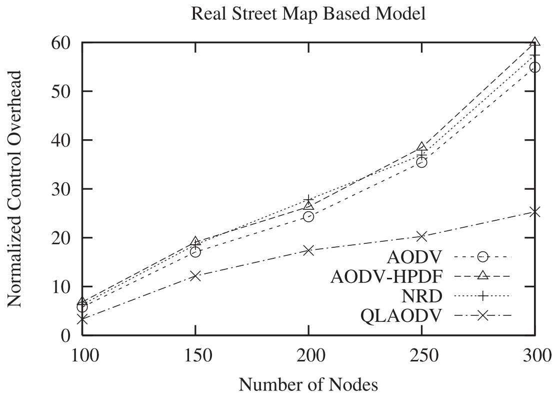


**Fig. 6** Normalized control overhead for varying number of nodes in real street map based mobility model.

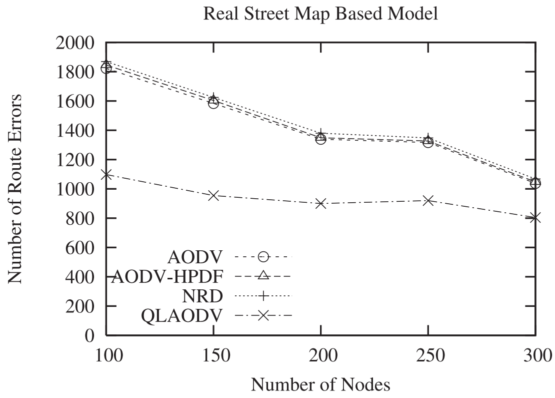


**Fig. 8** Number of route errors for varying number of nodes in real street map based mobility model.

tiate route discovery. Clearly, this introduces a high overhead. Although the mechanism of NRD for salvaging packets during mobility-initiated link breaks can avoid redundant route requests, unlike AODV-HPDF, it leads to non-optimal routes and therefore can not provide a significant improvement. Fortunately, the efficient route change mechanism in QLAODV reduces the number of route errors and therefore results in a low control overhead. As shown in Fig. 6, the normalized routing overheads of AODV, AODV-HPDF and NRD increase drastically with increasing node density. This is because the protocols use broadcast route discovery when a link failure occurs, which introduces a high overhead in a high-density network. Since QLAODV uses a unicast route change request/reply cycle to discover new routes, it results in a lower overhead.

Figure 7 and Fig. 8 show the number of route errors resulting from the four protocols. It is obvious that a dynamic route change mechanism results in a reduction in the number of route errors. In order to allow Q-Learning to work efficiently in a highly dynamic network environment, the QLAODV protocol uses additional packets, namely the route change request (RCNG-REQ) packet and route change replay (RCNG-REP) packet, to check the availability of candidate routes. Nevertheless, the RCNG-REQ packet and the RCNG-REP packet are sent unicast, and therefore this does not incur too great a network overhead.

In order to illustrate the efficiency of the route change mechanism with respect to varying velocity, we show the number of RCNG-REQ packets sent by source nodes and RCNG-REP packets received by source nodes in Fig. 9 and Fig. 10. Error bars indicate the standard deviation. A route change attempt fails if the source node of the RCNG-REQ packet does not receive the corresponding RCNG-REP. In the Manhattan model, we can see that many route change attempts fail when the velocity is high. This can be explained by the fact that high node velocity results in frequent topology changes and breakage of the candidate routes, which results in route change failure. In the freeway model, the occurrence of route change failures is not influenced much by the speed of movement because the relative speed between vehicles moving in the same direction would not be very high. In the real street map based model, when the node density increases, the number of route change requests increases slightly. This is because the number of available paths increases. However, we can also observe that the number decreases when the number of nodes increases further. This is because when the number of nodes increases, the average moving speed of vehicles will become slower.

As Fig. 11 and Fig. 12 show, the end-to-end delay of AODV-HPDF and NRD is larger than that of AODV and QLAODV. This is not a surprise since AODV-HPDF and NRD have longer route lengths and hence higher delays when compared to AODV. We also observe that QLAODV can construct shorter routes than AODV and thus can pro-

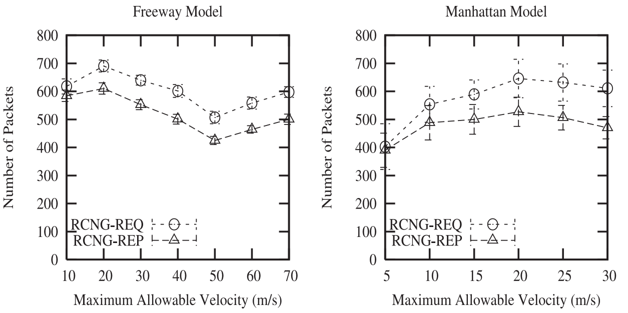


**Fig. 9** Number of RCNG-REQ packets sent by and RCNG-REP packets received by source node for varying velocities in freeway model (left) and Manhattan model (right).

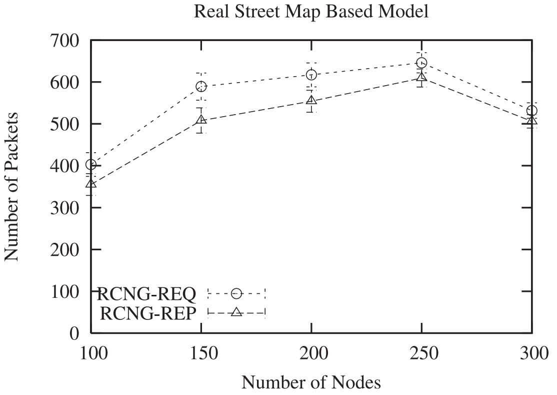


**Fig. 10** Number of RCNG-REQ packets sent by and RCNG-REP packets received by source node for varying number of nodes in real street map based mobility model.

vide a lower delay than AODV. To give a numerical proof to this behavior, we show the route length comparison of the four protocols in Fig. 13 and Fig. 14. Another reason why QLAODV achieves a good delay performance is that QLAODV reduces the number of route errors and route request broadcasts and so shortens the time packets are waiting in buffers. As QLAODV results in a lower delay than the other three protocols, it is suitable for use in multimedia applications and even in delay sensitive applications such as VoIP. It is clear from Fig. 13 and Fig. 14 that QLAODV can construct shorter routes than AODV. AODV-HPDF results in a longer route length due to the local repair without the hop-distance condition. NRD can salvage many packets in high-speed scenarios, but it results in a longer route.

## 4.2 Effect of the Transmission Rate

Figure 15 and Fig. 16 show the achieved packet delivery ratio, comparing the four protocols for varying transmission rate. In the freeway model, the maximum allowable vehicle velocity was 40m/s. In the Manhattan model, the maximum allowable velocity was set to 25m/s. We used 200 nodes in the real street map based mobility model. We generated 30 random CBR connections and simulated varying the transmission rate of each individual connection from 16 kbps to

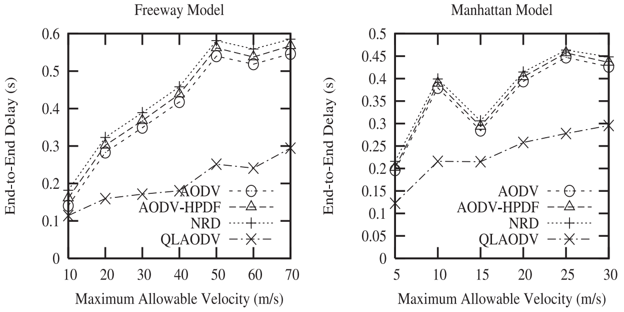


**Fig. 11** End-to-end delay for varying velocities in freeway model (left) and Manhattan model (right).

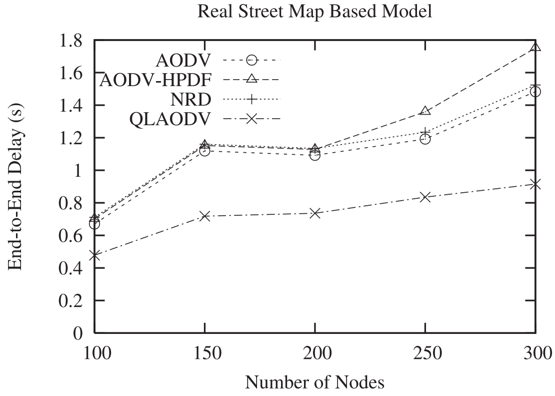


**Fig. 12** End-to-end delay for varying number of nodes in real street map based mobility model.

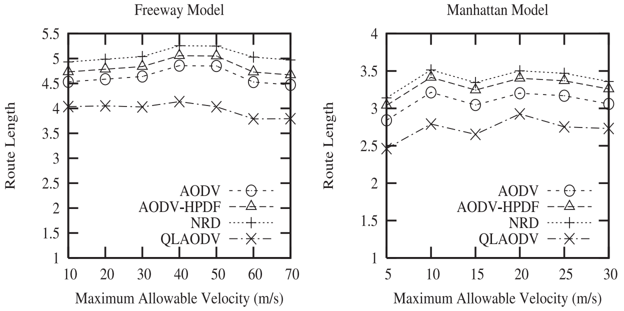


**Fig. 13** Route length for varying velocities in freeway model (left) and Manhattan model (right).

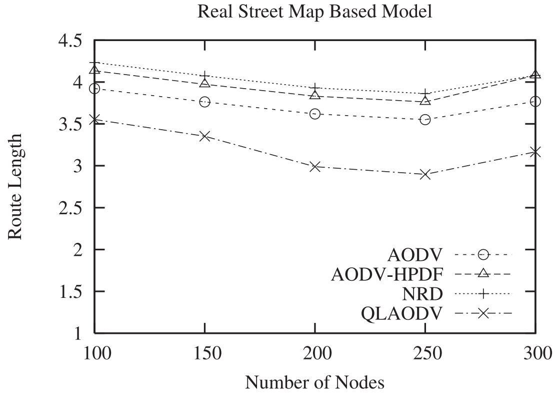


**Fig. 14** Route length for varying number of nodes in real street map based mobility model.

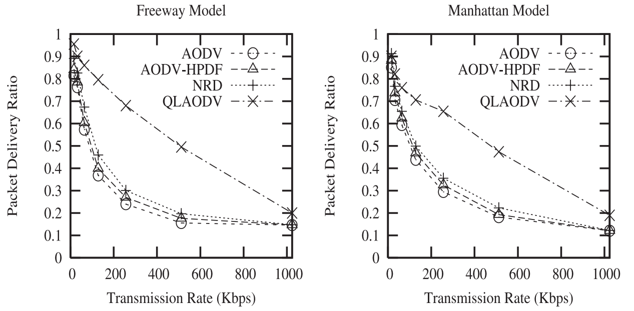


**Fig. 15** Achieved packet delivery ratio for varying transmission rates in freeway model (left) and Manhattan model (right).

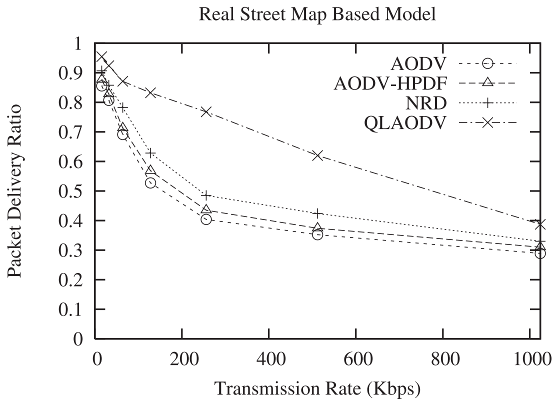


**Fig. 16** Achieved packet delivery ratio for varying transmission rates in real street map based mobility model.

1024 kbps.

It is clear from Fig. 15 and Fig. 16 that an increase of transmission rate results in an obvious negative impact on the packet delivery ratio of AODV. This is because the high transmission rate increases channel competition and network collisions. When the date rate is high, the number of data packets dropped upon link failure also increases. AODV-HPDF encounters same problem because of its high overhead. NRD only salvages packets that are dropped due to mobility, and not those dropped due to congestion. As the drops due to congestion become dominant, the NRD mechanism cannot make a significant positive impact on the overall performance. Since QLAODV considers bandwidth efficiency in the selection of the next hop and reduces the control overhead using a dynamic route change mechanism, it is superior to the other three protocols irrespective of the transmission rate.

## 4.3 Discussion

In this paper, we have provided extensive performance evaluations with different mobility models. In the freeway model, the moving speeds of vehicles can be very high. However, the relative speed between vehicles moving in the same direction may not be very high. Therefore QLAODV benefited from using vehicles moving in the same direction as the source node to forward data. In the Manhattan model and the real street map based mobility model, vehicles have freedom of changing moving direction at an intersection, it results in frequent link changes even when the vehicles' moving velocity is low. By preemptively changing routes before they break, QLAODV can achieve good performance.

In AODV, a node can use link layer notification or hello messages to keep track of its continued connectivity to its active next hop nodes. In this paper, we provide experimental results based on the assumption that link layer notification is available. In cases where the link layer notification is unavailable, AODV uses hello messages and AODV's performance drops drastically with increasing node velocity. In that case, the advantage of QLAODV is more apparent. This is because many packets would be dropped because AODV can not detect link failure quickly enough. Similarly, AODV-HPDF and NRD also face this problem. Thanks to its dynamic route change mechanism, QLAODV can handle this because it can switch to a new route before a link break occurs.

QLAODV also uses hello messages, to exchange link information. Nevertheless, this will not significantly impair the advantage of QLAODV because the messages are sent only periodically. In QLAODV, the hello interval is 1 s, so it will not incur too great an overhead compared to the route request broadcast of AODV in highly dynamic networks. It is also quite reasonable to use hello messages because it is necessary for every vehicle to be aware of its neighbors in a VANET. The simulation results confirm that QLAODV offers a significant performance improvement.

In QLAODV, every node has to maintain a Q-Table, which will consume more memory than the original AODV. However, this is not a problem in vehicular ad hoc networks because vehicles can have enough memory. Another concern is the size of hello messages. In QLAODV, every node attaches its MaxQValues to the hello messages to share its link state information with neighbors. The maximum number of elements in the MaxQValues can be equal to the number of nodes in the network. As the number of nodes increases, the information to be attached to the hello messages also increases, resulting in a higher message overhead. However, we can define a threshold value to control this overhead. An agent attaches a Q-Value to the hello messages only if its value is larger than the threshold because a smaller value would mean an inefficient path.

As described above, AODV-HPDF utilizes a local repair method in which both the upstream node and the source node initiate a route discovery when a link fails. While providing slightly better performance, this mechanism results in high control overheads during situations of high mobility. In NRD, when a link on a route fails due to mobility, the intermediate node on the failed link forwards packets to one of its neighbors to which the route information for the corresponding destination has already been diffused. NRD salvages packets efficiently in the case of multiple streams terminating at a single destination node. However, as the streams become more distributed, NRD's effectiveness diminishes. Moreover, the NRD mechanism can not make a significant improvement when the packet drops due to congestion, as opposed to link failure, become dominant. Fortunately, QLAODV can offer a notable performance improvement in various situation. First, the novel dynamic route change mechanism is more effective than taking action after link failure. Another merit of QLAODV is that it considers hop count, stability and bandwidth efficiency in route selection, making QLAODV very robust to network dynamics.

## 5. Conclusions

In this paper, we have proposed QLAODV, a routing protocol that uses a reinforcement learning algorithm to handle network state information and a unicast route change request/reply cycle to check the correctness of the information obtained. QLAODV uses a dynamic route change mechanism to reduce the number of route errors and route discoveries. QLAODV can react quickly to network topology changes and can pick the best route for data delivery using newly learned information. QLAODV considers hop count, stability and bandwidth usage in route selection. It is a fully topology-based routing protocol and is therefore easy to implement. Through exhaustive evaluation of the proposed routing protocol on different mobility models, we have confirmed that QLAODV offers a significant performance advantage over existing alternatives.

### References

[1] S. Chaabouni, M. Frikha, and M. Meincke, "Traffic models for inter-vehicle communications," Proc. Second International Conf. on Inf. and Commun. Technologies, pp.773–778, Damascus, Syria, April 2006.

[2] C.V. Jasmine, C. Wai, A. Onur, and C. Shengwei, "Survey of routing protocols for inter-vehicle communications," Proc. 3rd Annual International Conf. on Mobile and Ubiquitous Systems, pp.1–5, San Jose, USA, July 2006.

[3] F. Li and Y. Wang, "Routing in vehicular ad hoc networks: A survey," Vehicular Technology Magazine, vol.2, no.2, pp.12–22, June 2007.

[4] M. Mauve, A. Widmer, and H. Hartenstein, "A survey on position-based routing in mobile ad hoc networks," IEEE Network Magazine, vol.15, no.6, pp.30–39, Dec. 2001.

[5] R. Jain, A. Puri, and R. Sengupta, "Geographical routing using partial information for wireless ad hoc networks," IEEE Pers. Commun., vol.8, no.1, pp.48–57, Feb. 2001.

[6] L.P. Kaelbling, M.L. Littman, and A.W. Moore, "Reinforcement learning: A survey," J. Artificial Intelligence Research, vol.4, pp.237–285, 1996.

[7] C. Watkins, Learning from Delayed Rewards, Ph.D. Thesis, King's College, Cambridge, 1989.

[8] B. Karp and H.T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," Proc. 6th Annual International Conf. on Mobile Computing and Networking, pp.243–254, New York, USA, 2000.

[9] M. Zhaomin, Z. Hao, M. Kia, and N. Pissinou, "MURU: A multi-hop routing protocol for urban vehicular ad hoc networks," Proc. 3rd Annual International Conf. on Mobile and Ubiquitous Systems, pp.1–8, San Jose, USA, July 2006.

[10] V. Namboodiri and L. Gao, "Prediction-based routing for vehicular ad hoc networks," IEEE Trans. Veh. Technol., vol.56, no.4, pp.2332–2345, 2007.

[11] T. Taleb, E. Sakhaee, A. Jamalipour, K. Hashimoto, N. Kato, and Y. Nemoto, "A stable routing protocol to support ITS services in VANET networks," IEEE Trans. Veh. Technol., vol.56, no.6, pp.3337–3347, Nov. 2007.

[12] Q. Yang, A. Lim, and P. Agrawal, "Connectivity aware routing in vehicular networks," Proc. IEEE Wireless Commun. and Networking Conf., pp.2218–2223, Las Vegas, USA, April 2008.

[13] B. Ducourthial, Y. Khaled, and M. Shawky, "Conditional transmissions: Performance study of a new communication strategy in VANET," IEEE Trans. Veh. Technol., vol.56, no.6, pp.3348–3357, Nov. 2007.

[14] G. Lu, G. Manson, and D. Belis, "Enhancing routing performance for inter-vehicle communication in city environment," Proc. ACM International Workshop on Performance Monitoring, Measurement, and Evaluation of Heterogeneous Wireless and Wired Networks, pp.82–89, Torremolinos, Spain, 2006.

[15] C. Perkins and E. Belding-Royer, Ad hoc On-Demand Distance Vector (AODV) Routing, RFC 3561, July 2003.

[16] H. Menouar, M. Lenardi, and F. Filali, "An intelligent movement-based routing for VANETs," ITS World Congress 2006, London, United Kingdom, Oct. 2006.

[17] W. Wang, F. Xie, and M. Chatterjee, "TOPO: Routing in large scale vehicular networks," Proc. 66th IEEE Vehicular Technology Conf., pp.2106–2110, Baltimore, USA, Oct. 2007.

[18] O. Abedi, M. Fathy, and J. Taghiloo, "Enhancing AODV routing protocol using mobility parameters in VANET," Proc. IEEE/ACS International Conf. on Computer Systems and Applications, pp.229–235, Doha, Qatar, April 2008.

[19] C. Liang and H. Wang, "An ad hoc on-demand routing protocol with high packet delivery fraction," Proc. IEEE International Conf. on Mobile Ad-hoc and Sensor Systems, pp.594–596, Fort Lauderdale, USA, Oct. 2004.

[20] M. Quwaider, J. Rao, and S. Biswas, "Neighborhood route diffusion for packet salvaging in networks with high mobility," Proc. IEEE International Conf. Performance, Computing and Communications, pp.168–175, Austin, USA, Dec. 2008.

[21] J.A. Boyan and M.L. Littman, "Packet routing in dynamically changing networks: A reinforcement learning approach," Advances in Neural Information Processing Systems, vol.6, pp.671–678, 1993.

[22] Y.H. Chang, T. Ho, and L.P. Kaelbling, "Mobilized ad-hoc networks: A reinforcement learning approach," Proc. First International Conf. on Autonomic Computing, pp.240–247, New York, USA, May 2004.

[23] J. Dowling, E. Curran, R. Cunningham, and V. Cahill, "Using feedback in collaborative reinforcement learning to adaptively optimize MANET routing," IEEE Trans. Syst. Man Cybern., vol.35, no.3, pp.360–372, May 2005.

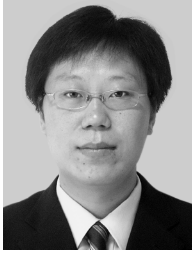
[24] R.D. Renesse, M. Ghassemian, V. Friderikos, and A.H. Aghvami, "Adaptive admission control for ad hoc and sensor networks providing quality of service," Technical Report, King's College London, 2005.

[25] C.J.C.H. Watkins and P. Dayan, "Q-Learning," Mach. Learn., vol.8, no.3-4, pp.279–292, 1992.

[26] F. Bai, N. Sadagopan, and A. Helmy, "Important: A framework to systematically analyze the impact of mobility on performance of routing protocols for adhoc networks," Proc. 22nd Annual Joint Conf. of the IEEE Computer and Commun. Societies, pp.825–835, San Francisco, USA, April 2003.

[27] U.S. Census Bureau — TIGER/Line, http://www.census.gov/geo/www/tiger/

[28] A.K. Saha and D.B. Johnson, "Modeling mobility for vehicular ad hoc networks," Proc. First ACM Workshop on Vehicular Ad Hoc Networks, pp.91–92, Philadelphia, USA, Oct. 2004.



**Celimuge Wu** received the M.E. degree from Beijing Institute of Technology, Beijing, China, in 2006. He is currently a Ph.D. candidate at Department of Information Network Science, Graduate School of Information Systems, the University of Electro-Communications, Tokyo, Japan. His current research interests include mobile ad hoc networks, networking architectures and protocols.

**Kazuya Kumekawa** received the B.S. degree in engineering, the M.S., and Ph.D. degrees in science from Tohoku University in Japan, in 1992, 1994, and 1997, respectively. He is currently an Assistant Professor at Department of Information Network Science, Graduate School of Information Systems, the University of Electro-Communications in Tokyo, Japan.

**Toshihiko Kato** received the B.E., M.E. and Dr.Eng. degrees electrical engineering from the University of Tokyo, in 1978, 1980 and 1983, respectively. He joined KDD in 1983 and worked in the field of communication protocols of OSI and Internet until 2002. From 1987 to 1988, he was a visiting scientist at Carnegie Mellon University. He is now a professor of the Graduate School of Information Systems in the University of Electro-Communications in Tokyo, Japan. His current research interests include protocol for mobile Internet, high speed Internet and ad hoc network.