QoS-aware roadside base station assisted routing in vehicular networks ☆

Chenn-Jung Huang [a,*], Yi-Ta Chuang [b], You-Jia Chen [a], Dian-Xiu Yang [a], I-Fan Chen [a]

[a] *Department of Computer & Information Science, College of Science, National Hualien University of Education, Hualien 970, Taiwan*
[b] *Institute of Computer Science and Engineering, National Chiao Tung University, Hsinchu 300, Taiwan*

ARTICLE INFO

ABSTRACT

The transmission technology for intelligent transportation systems can be typically classified into two categories, namely, road-to-vehicle communication (RVC) and inter-vehicle communication (IVC). RVCs perform the information communication service offer from road to vehicle whereas the IVCs perform the information communication through vehicles. This work proposes quality of service (QoS)-aware roadside base station assisted routing mechanisms to establish a routing path in IVC with the assistance of roadside base station. A link failure prevention mechanism is employed to effectively construct alternative routing path required by the volatile network topology in vehicular Ad hoc networks. Besides, a bandwidth consumption predictor is presented to avoid dropping packets owing to inadequate bandwidth during handoffs. A neural network with fast learning algorithm is adopted as the core module for estimating the parameters used in the proposed schemes. Simulation results demonstrate the effectiveness and feasibility of the proposed work.

1. Introduction

In recent years, the research on the intelligent transportation systems (ITSs) has been progressing intensively. ITS uses information communication technologies to connect "the person", "the road", and "the vehicle" as one system. The transmission technology for ITS can be typically classified into two categories, i.e., road-to-vehicle communications (RVC) and inter-vehicle communications (IVC). IVCs are achieved using effective routing protocol that considers the specific characteristic of the road information, relative car movements and application restriction. The most important requirement is the quality of service (QoS), especially the communication delay between the end-to-end and the minimum consumption of network resources. Wireless mobile Ad hoc network (MANET) technologies promise delivery of network access area without the need of infrastructure, which is required by other technologies. However, MANET technologies cannot be directly applied to IVCs since the characteristics of vehicle movement and relative speed of mobile nodes are different from those in traditional Ad hoc network. There have been several researches (Mesh Networks Inc., 2000; Cherry, 2003) addressed on the construction of Ad hoc network among vehicles in the early stage of development of MANETs. Recently, the usage

of MANETs as a base technology in IVCs has gained more popularity due to its potential applications, such as providing support for intelligent transportation systems (ITS) and expediting Internet access in high ways.

It is well known that the major challenge for designing routing protocols in MANETs is to find a path from the source to the destination without any preconfigured information or regularly varying link situations. The position-based routing becomes a suitable candidate for vehicular Ad hoc networks (VANET) because this kind of routing protocol depends only on the geographic position information and the information can be easily obtained by navigation systems, such as global position system (GPS) (Maihofer and Eberhardt, 2004; Imielinski and Navas, 1996). In addition, the dissemination of the network and the road information can be more efficient if base station allocated for RVCs can be arranged to participate in the determination of management policy or routing path construction in IVCs and RVCs.

This work presents roadside base station assisted routing mechanisms, which are able to dynamically choose the routing path through multi-hop vehicular Ad hoc networks or roadside base station according to the current network conditions. Link break and congestion indicator mechanisms are proposed to ease the link break or congestion situation that might occur in IVCs. Meanwhile, a bandwidth consumption predictor is presented to estimate the consumed bandwidth for each roadside base station in RVCs during the next period of time. Notably, an advanced neural-network architecture that was recently introduced in the literature, so-called fast learning neural networks (FLNNs)

(Zhu et al., 2004), is employed to derive the numeric values for the parameters needed in the above mentioned routing mechanisms. Unlike traditional neural networks, the FLNNs not only produce good generalization performance but also have real-time learning and prediction capability. To the best of our knowledge, this work is the first application of the FLNNs to the network routing problems. The experimental results exhibit the feasibility and practicality of our proposed work.

The rest of this paper is organized as follows. Section 2 gives a brief overview of the IVCs and RVCs in the literature. Section 3 presents the proposed QoS-aware roadside base station assisted routing mechanisms. The simulation results and analysis are given in Section 4. Conclusion is made in Section 5.

2. Related work

The researches on IVCs can be roughly divided into three categories: unicast, flooding, and diffusion in the literature. Traditional Ad hoc network routing protocols (Füssler et al., 2002) or position-based routing protocols (Mauve et al., 2001; Wu et al., 2004) can be used to establish general unicast communication in a VANET. A service discovery mechanism is then established to allow each node to obtain the needed information (Festag et al., 2004; Morris et al., 2000). Nevertheless, the overhead such as the latency and the diminished network capacity caused by the service discovery mechanism and routing table maintenance makes this method infeasible for most safety critical applications.

The flooding and diffusion rely on the observation that the importance of sensed information about a particular location decreases with the distance to that location. Data is thus required to be disseminated in the vicinity of its origin. This is the case for most safety applications, but not for example infotainment (Bogenberger et al., 2003) or environmental applications, where all data comes from some remote site(s). Most IVC protocols employ flooding to broadcast data, in which the performance drops quickly as the number of nodes increases because each node receives and broadcasts the message simultaneously and contentions and collisions, broadcast storms and high bandwidth consumption might occur (Ni et al., 1999). A so-called dedicated omni-purpose inter-vehicle communication linkage protocol for highway automation (DOLPHIN) (Tokuda et al., 2000) is an example of IVC in this category. Selective flooding is used to disseminate the information in the reverse direction of vehicle movement. The nodes that broadcast the information are reselected in every communication hop and would not have a good performance for point-to-point communications. The GPS-based message broadcasting (Sun et al., 2000) uses a better broadcasting system, similar to the single cast routing protocol, zone routing protocol (ZRP). It performs much better than flooding based ones, but it still has routing overhead as long as the forwarding nodes are selected in every hop and is not efficient for point-to-point communications.

Korkmaz et al. (2006) proposed a cross-layer protocol using clustering transmission (CVIA). They create single-hop vehicle clusters and mitigate the hidden node problem by dividing road into segments and controlling the active time of each segment. However, the assumption of each vehicle on the road moves at a fixed speed without considering the impact of mobility causes this approach infeasible in the application of VANETs.

Unlike all of the solutions mentioned above, this work exploits roadside base station assisted routing mechanisms that adapt to the architecture of IVC/RVC and the specific characteristic of VANETs, and tackles the unresolved issues mentioned in the above brief discussions on the related work.

3. Roadside base station assisted routing mechanisms for VANETs

3.1. Routing path construction

In this work, the robust communications in the VANETs are established by constructing an effective routing path on which the vehicles can transmit or receive their packet through IVC or RVC. As shown in Fig. 1, the base station is employed to determine the routing paths for the vehicles on the road segment that the base station governs. When a vehicle enters the road segment and submits a packet transmission request, the administrative base station attempts to arrange a shortest routing path to the destination for the packets via IVC if the traffic in IVC is not congested. The IVC protocol adopted in this work is similar to the approach taken in the AODV, in which an acknowledgment (ACK) packet is sent back to the source node when the destination node receives a packet in order to certify that each packet is successfully delivered. In case no appropriate routing path can be found in IVC, the administrative base station can grant the vehicle's request when there is enough free bandwidth to meet the request's minimum bandwidth requirement. Notably, the reason for selecting AODV-like protocol in this work is that the route discovery mechanism in a connection-oriented protocol, such as AODV, precisely matches the bandwidth calculation scheme and is suitable for bandwidth constrained routing. In addition, AODV provides some minimal control to enable nodes to specify quality of service parameters, namely maximal delay or minimal bandwidth, that a route to a destination must satisfy (Chen and Nahrstedt 1999).

The traffic transmitted in this work is classified as either real-time traffic or non-real-time traffic. The freeable bandwidth for the request of real-time traffic can be expressed by

$$Bw_f = Bw_{unused} + \sum_i (Bw_{i,curr} - Bw_{i,min}) - Bw_e, \tag{1}$$

where Bw_{unused} is the unused bandwidth at the base station, $Bw_{i,curr}$ is the bandwidth currently allocated for the non-real-time traffic with index i, $Bw_{i,min}$ is the minimum bandwidth required for the non-real-time traffic with index i, and Bw_e is the bandwidth reserved for transmission of emergency events.

As for the non-real-time traffic, the so-called freeable bandwidth is exactly the unallocated bandwidth at the base station, Bw_{unused}. Notably, code-division multiple-access (CDMA) is adopted in this work to ease the interference effect and increase the transmission efficiency as in (Souryar et al., 2006; Kawamagari et al., 2006; Elbahhar et al., 2001; Linedemmeier et al., 2003; Chung and Cho, 2006). Besides, the administrative base station will direct a route via the roadside base station that is closer to the destination, and all the roadside base stations will follow the same decision procedure as illustrated in Fig. 1 in case they receive the incoming requests from other base stations.

3.2. Link enhancement mechanism for IVC via RVC

In the VANETs, the fault-tolerant connectivity can be established by offering alternative routing paths whenever a possible link failure or congestion event occurs on the current routing path. We thus arrange each node in IVC to compute link break and congestion indicators to avoid possible link break and congestion events occurring at each node. Each vehicle will inform its administrative base station in case link break and congestion events are anticipated. The base station will then look up its routing table to construct the alternative route. Notably, the base station keeps monitoring the network status of each node on the
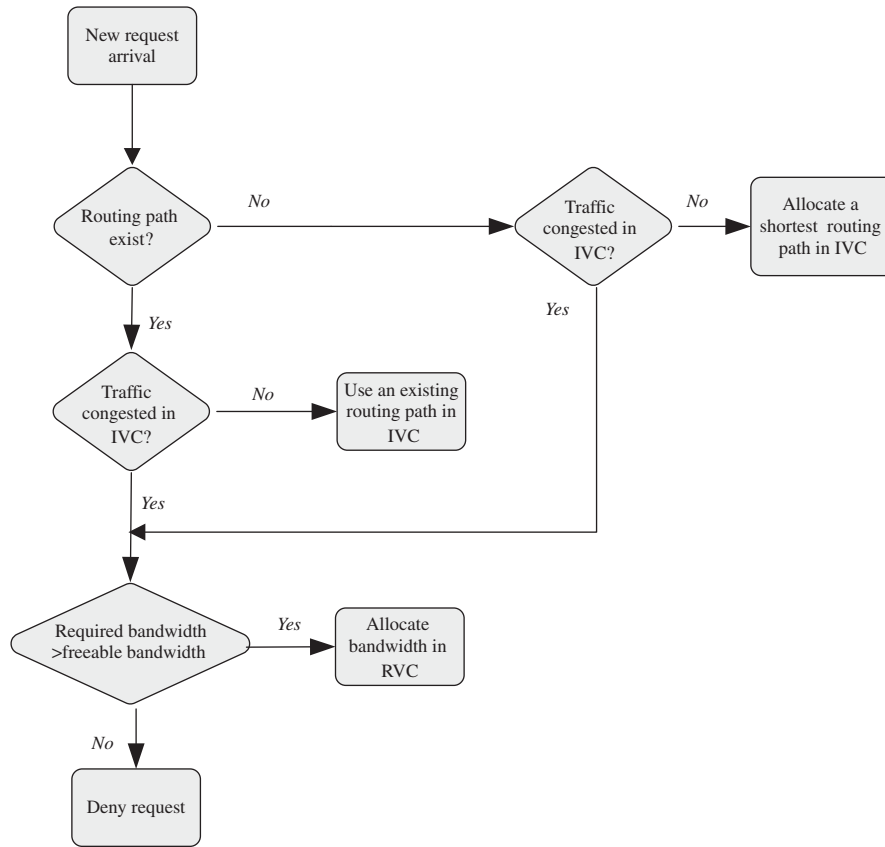
Fig. 1. The proposed roadside base station assisted routing mechanism.

road segment that it governs, and shares the information with the neighboring roadside base stations.

3.2.1. Link failure avoidance based on link break and congestion indicators

In order to prevent link break caused by varied mobility or congestion events, we attempt to estimate each vehicle's speed in the next time period using a neural network with fast computing capability. The input–output mapping for the employed neural network can be expressed by

$$v_{next} = \mathrm{NN}(d, a, r, v_{cur}), \tag{2}$$

where d denotes the distance between two consecutive vehicles, a represents the driver's age, r is condition of the road segment, and v_{cur} is the current speed of the vehicle.

Notably, the driver's age is adopted as one of the parameters for the prediction of vehicle speed because the influence of driver's age on his/her driving behavior has been an active research issue in recent year (de Ridder et al., 2002; Sivak et al., 1981; DeLucia et al., 2003). On the contrary, other factors, such as "wearing glasses" and "weather", have never been reported to influence the driving behavior in the literature to the best our knowledge. It was observed in de Ridder et al. (2002) that older participants made more mistakes than did younger participants in both real and simulated driving tasks. Additionally, older drivers require closer distances to correctly perceive the orientation of the letter on the nighttime highway sign (Sivak et al., 1981). Older participants overestimate speed at lower velocities, underestimate speed at higher velocities, and underestimate time-to-contact more than younger drivers (DeLucia et al., 2003).

Once the predicted speeds of the vehicle and its neighbors are obtained, we can easily determine whether the vehicle is within the communication range of its neighbors by computing the distances of the vehicle and its neighbors in the next time period as follows:

$$p_{next} = v_{next} + p_{cur}, \tag{3}$$

where v_{next} denotes the speed of the vehicle in the next measuring period and p_{cur} is the current position of the vehicle.

In case the vehicle's position is estimated to be out of the communication of its neighbors in the next time period, the vehicle can initiate backup route construction process to prevent link failure caused by mobility of vehicles by piggybacking link break warning message to its neighbors.

Similar to Eq. (2), the congestion indicator can be derived by

$$cg = \mathrm{NN}(q, p, n), \tag{4}$$

where q denotes the queue length, p is the expected number of the packets traveling through the vehicles, n is the expected number of the vehicles in the next time period, and the function NN stands for the proposed neural-network model.

Now take Fig. 2 as an example, when there is a possible congestion or link break detected at node B, it sends a congestion/link break warning message to all its neighbors. As node A receives the message, it re-initiates route discovery process with congestion/link break indicator piggybacked in packets to find an alternate path to destination D. Thus, new arrived packets can then be delivered via a new path as shown in Fig. 2.

3.3. Bandwidth consumption predictor for roadside base station

In this work, the routing path for transmitting packets is either through IVC or RVC. There should be a bandwidth management mechanism because the roadside base station has limited bandwidth. The estimated position of the moving vehicle in the next time period given by Eq. (3) can be used to count the number
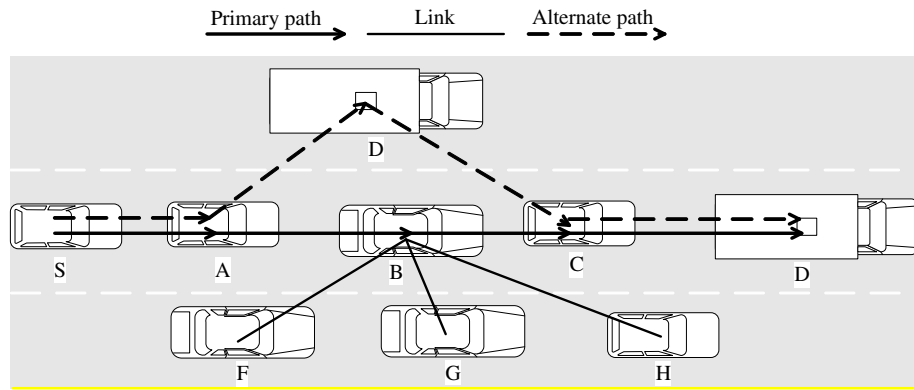
Fig. 2. Alternate path construction.

of vehicles that are located at the road segment that the roadside base station governs. The bandwidth consumed by the vehicles moving on each road segment can then be computed by

$$Bw_{used} = \sum_i b_{i,incoming} + \sum_j b_{j,cur} - \sum_k b_{k,outgoing}, \qquad (5)$$

where $b_{i,incoming}$, $b_{j,cur}$ and $b_{k,outgoing}$ represent the required bandwidth for the ith vehicle that is moving into the target road segment, the jth vehicle that stays on the target road segment, and the kth outgoing vehicle in the next time period, respectively.

The roadside base station that is expected to run out of bandwidth during the next time period will inform its neighboring roadside base stations that it is unable to receive new routing requests for the time being due to scarce bandwidth.

3.4. Fast learning neural networks

As given in Eqs. (2) and (4), a recently proposed neural-network architecture, fast learning neural networks (FLNNs), is employed as the core module in the computation of link break and congestion indicators. The neural networks are chosen owing to their superior performance in the literature (Huang et al., 2006). The major problem of applying neural networks is that it is not impractical to use neural networks in real-time applications because it takes too much time during the neural-network training process. However, there have been lots of solutions on VLSI chips that allow the neural networks to be hardware-computed; thus, the implementation of neural networks by hardware becomes feasible nowadays. Besides, this work replaces the classical back-propagation learning by a fast learning algorithm (FLA) to reduce the overlong training time that usually occurs in the traditional neural-network models. The training process for the neural networks is activated at each node whenever the performance metric such as packet delivery ratio of the traffic degrades below some preset threshold.

The FLNNs (Huang et al., 2006, 2000; Huang, 2003; Zhu et al., 2004) that adopt the fast learning algorithm can automatically select appropriate value of neural quantizers and analytically determine the weights and bias of the network at one time. As shown in Fig. 3, a FLNN is composed of complex network structure. The FLNN employs a powerful and novel constructive method to remarkably reduce the computing time of the neural networks, especially for some problems with large training set. It is possible for the FLNN to represent all observations with arbitrarily high accuracy and make it feasible to implement extensive systems in an ordinary computer. Xiang et al. (2005)

observed that four-layered neural networks are more powerful than three-layered neural networks in most cases. For instance, one obvious advantage gained by decomposing the target function into several sub-functions is that the total number of the parameters of the four-layered MLP may be smaller than that of three-layered MLP. Obradovic and Yan (1990) reported the superiority of four-layered feed-forward networks to three-layered networks in terms of mapping capabilities. They showed that the classification boundaries of four-layered networks, which are at most polynomial in the number of training samples, are strictly more general than those of three-layered networks. Based on the previous analysis in the literature, Huang took advantage of using four layers in the design of the neural networks and presented a constructive network and a real-time learning algorithm for the architecture. Based upon Huang's constructive method, the connection weights and bias linking the first hidden layer and second hidden layer can be determined at one time, instead of the iterative adjustment method adopted in most neural-network learning algorithms. Hence, the real-time learning algorithm can automatically select appropriate values of neural quantizers and analytically determine the weights and bias of the network at one time only. This kind of neural networks has the following characteristics:

(1) The neural networks are composed of two hidden layers, with first larger and second narrower. There are several neuron quantizers that connect to the input layer and one or several neurons in the second hidden layer.
(2) All the connection weights between the input layer and the first hidden layer can be simply prefixed, and most of them are randomly assigned.
(3) The connection weights between the first hidden layer and second hidden layer can be determined at one time, instead of the iterative adjustment adopted in traditional neural-network learning algorithms.
(4) The connection weights between the second hidden layer and the output layer can be simply set as a constant value C.
(5) Weight size factor wf or C and quantizer factors qt and qu should be adjusted to optimize generalization performance.

The fast learning algorithm adopted in this work can be summarized as follows:

Inputs of FLA:

1. N arbitrary distinct samples (x_i, t_i), where $x_i = [x_{i1}, x_{i2}, \ldots, x_{in}]^T \in \Re^n$ and $i = 1, \ldots, N$.
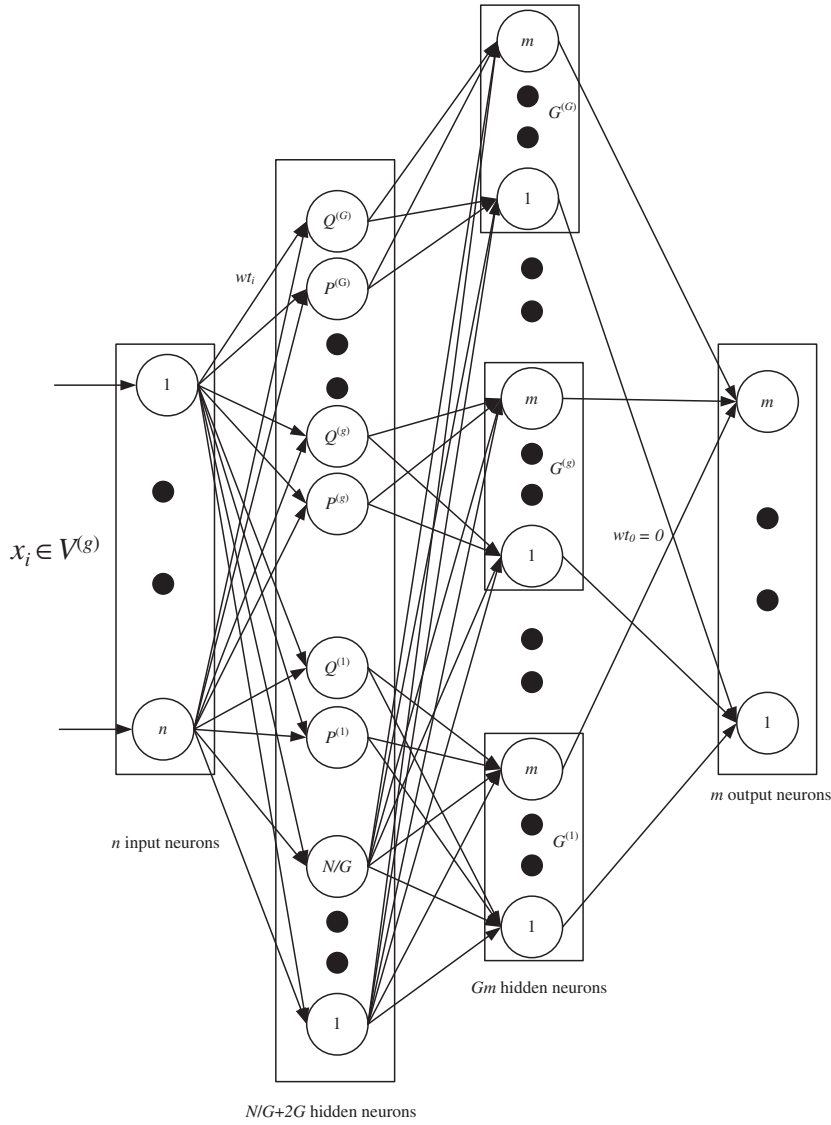
Fig. 3. Neural network with fast learning algorithm.

2. Expected learning accuracy $\varepsilon < 0$.
3. Number of groups G.
4. Number of neurons D of the first hidden layer.

The learning procedure of FLA:
  Step 1: Sorting and grouping inputs.

(a) Randomly choose vector $wt \in \Re^n$ and re-index inputs such that $wt \cdot x_1 < wt \cdot x_2 < \cdots < wt \cdot x_N$.
(b) Group sorted inputs into G groups $V^{(g)}$, $g = 1 \sim G$

$$V^{(g)} = \{x_i | wt \cdot x_{(g-1)N/G+1} \leqslant wt \cdot x_i \leqslant wt \cdot x_{gN/G}\}. \tag{6}$$

Step 2: Determination of weights and biases of two hidden layers.

(a) Randomly choose the weights wt_i and biases b_i, where $i = 1,\ldots,D$, and $D \leqslant N/G$ is the number of neurons in the first hidden layer.
(b) Choose $C = \alpha \cdot \max_{1 \leqslant i \leqslant n; 1 \leqslant j \leqslant m} |t_{ij}|$, where $\acute{a}$ can be any positive number larger than two.

(c) Calculate matrix $\gamma^{(g)}[\gamma_1^{(g)}, \gamma_2^{(g)}, \ldots, \gamma_D^{(g)}]^T$:

$$\gamma^{(g)} = (M^{(g)})^* U^{(g)} \gamma^{(g)} = (M^{(g)}), \tag{7}$$

where

$$M^{(g)} = \begin{bmatrix} \sigma(wt_1 \cdot x_{(g-1)N/G+1} + b_1) & \ldots & \sigma(wt_D \cdot x_{(g-1)N/G+1} + b_D) \\ \vdots & \ldots & \vdots \\ \sigma(wt_1 \cdot x_{gN/G} + b_1) & \ldots & \sigma(wt_D \cdot x_{gN/G} + b_D) \end{bmatrix}, \tag{8}$$

$$\sigma(x) = 1/(1 + e^{-x}),$$

$$U^{(g)} = \begin{bmatrix} \ln\left(\dfrac{0.5 + t_{(g-1)N/G+1,1}/C}{0.5 - t_{(g-1)N/G+1,1}/C}\right) & \ldots & \ln\left(\dfrac{0.5 + t_{(g-1)N/G+1,m}/C}{0.5 - t_{(g-1)N/G+1,m}/C}\right) \\ \vdots & \ldots & \vdots \\ \ln\left(\dfrac{0.5 + t_{gN/G,1}/C}{0.5 - t_{gN/G,1}/C}\right) & \ldots & \ln\left(\dfrac{0.5 + t_{gN/G,m}/C}{0.5 - t_{gN/G,m}/C}\right) \end{bmatrix}, \tag{9}$$

and $(M^{(g)})^*$ stands for the Moore–Penrose generalized inverse of $(M^{(g)})$.

Step 3: Determination of weights and biases of neural quantizers.

(a) Set the quantizers factor qt and qu as

$$qt = \frac{2\ln(2qu/\min_{i=1\sim N}\ln((C + \varepsilon/\sqrt{m} - 2t_{ij})(C + 2t_{ij})/(C - \varepsilon/\sqrt{m} + 2t_{ij})(C - 2t_{ij})) - 1)}{\min_{j=1\sim N-1}(wt \cdot x_{j+1} - wt \cdot x_j)}, \tag{10}$$

$$qu = \ln\left(\frac{2\sqrt{m}CG}{\varepsilon} - 1\right) + \max_{\substack{g=1\sim G \\ q=1\sim N/G \\ s=1\sim G}} \|M_q^{(g)} \cdot \gamma^s\|_\infty$$

$$+ \min_{i=1\sim N}\ln\left(\frac{(C + \varepsilon/\sqrt{m} - 2t_{ij})(C + 2t_{ij})}{(C - \varepsilon/\sqrt{m} + 2t_{ij})(C - 2t_{ij})}\right). \tag{11}$$

(b) Set the connection weights $wt_{P^{(g)}}$ and $wt_{Q^{(g)}}$ that link the input layer and neurons $P^{(g)}$ and $Q^{(g)}$, $g = 1, 2, \ldots, G$, as

$$wt_{P^{(g)}} = qt \cdot wt$$
$$wt_{Q^{(g)}} = -qt \cdot wt, \tag{12}$$

(c) Set the biases of neurons $P^{(g)}$ and $Q^{(g)}$, $g = 1, 2, \ldots, G$ as

$$\bar{b}_{P^{(g)}} = \begin{cases} -qt\left(\frac{1}{2}wt \cdot x_{gN/G} + \frac{1}{2}wt \cdot x_{gN/G+1}\right), & \text{if } g \neq G \\ -qt\left(wt \cdot x_N + \max_{j=1\sim N-1}(wt \cdot x_{j+1} - wt \cdot x_j)\right), & \text{if } g = G \end{cases} \tag{13}$$

$$\bar{b}_{Q^{(g)}} = \begin{cases} qt\left(\frac{1}{2}wt \cdot x_{(g-1)N/G+1} + \frac{1}{2}wt \cdot x_{(g-1)N/G}\right), & \text{if } g \neq 1 \\ qt\left(wt \cdot x_1 + \max_{j=1\sim N-1}(wt \cdot x_{j+1} - wt \cdot x_j)\right), & \text{if } g = 1 \end{cases} \tag{14}$$

(d) Set the connection weights w_{AB} that link neurons $A^{(p)}$ and $A^{(p)}$ and the second hidden layer as

$$w_{PQ} = -qu \tag{15}$$

(e) Set the connection weights and the biases that link neurons of the second hidden layer and the output layer as

$$wt_o = C \tag{16}$$

$$\bar{b}_o = -0.5C \tag{17}$$

4. Experimental results

We ran a series of simulations to evaluate the performance of the proposed work using a network simulator written by C++. The results are averaged over 130 runs with a randomly generated topology in each run. The total simulation time for each run is 500 s. The compared schemes include the proposed roadside base station assisted routing algorithm without link enhancement mechanism (BAR), the proposed roadside base station assisted routing mechanisms embedded with fast learning neural-network link break and congestion detection modules (BAR-FLNNLC), the roadside base station assisted routing mechanisms embedded with back-propagation neural-network link break and congestion detection modules (BAR-BPNNLC) and a representative cross-layer

routing protocol for ad hoc networks that was recently introduced in the literature (CVIA) (Korkmaz et al., 2006). CIVA creates single-hop vehicle clusters and mitigates the hidden node problem by dividing road into segments and controlling the active segments where vehicle communication is allowed to occur in a time slot.

4.1. Simulation scenario

The highway and local road simulation environment setups were replicated from (Namboodiri and Gao, 2007; Zhao et al., 2007), respectively. A $5000\,\text{m} \times 5000\,\text{m}$ rectangle area was established in the local roads simulation. We initially randomly deployed vehicles on local roads. Each vehicle randomly picked a value as its moving speed and 20 m/s was the upper speed limit. An 8000-m-long straight stretch of highway with four lanes was constructed as the high-speed moving environment. Fifty vehicles were randomly distributed on the highway. Based on the lane in which the vehicles were randomly placed, all vehicles were given an initial velocity. The initial velocity assigned to each vehicle was the average speed of the lane in which they were placed in. In order to simulate the road traffic, the traffic flow is simulated with microscopic model (van Arem et al., 2006). The detail simulation parameters are listed in Table 1.

Notably, CBR/UDP traffic is generated between randomly selected pairs of vehicles and the bandwidth for each channel is 2 Mbps. The bandwidth of the base station is 54 Mbps, there is one base station located in each road segment. There are two service classes, including real-time and non non-real-time traffics. The CBR packet size is 512 byte and packet rate is 4 packets/s. Each vehicle moves along the direction of the pathway, and the speed is randomly changed within a preset range that is related to the driver's age and the distance between the vehicle and the one in front of it. Once it reaches that position, it will change the speed and repeats the process.

4.2. Simulation results and analysis

The performance metrics used in this work are packet delivery ratio, end-to-end delay and control overhead. The packet delivery

Table 1
Simulation parameters.

Parameter type	Parameter value
Simulation time	500 s
Rectangle area of local road	$5000\,\text{m} \times 5000\,\text{m}$
Length of highway	8000 m
Length of the road segment	1000 m
Traffic flow	0.1–0.5 veh/s
Traffic model	Microscopic model
Average vehicle speed	5–30 m/s
Maximum vehicle speed	50 m/s
Channel bandwidth	2 Mbps
Mac protocol	802.11
Transmission range for IVC	33.75 m
Transmission range for RVC	250 m
Bandwidth of base station	54 Mbps
Service class	Real-time, non-real-time
CBR real-time sessions	25

ratio is the total amount of received packets divided by the total amount of packets transmitted during the simulation. The end-to-end delay is the average time elapsed for delivering a packet from the transmitter to the receiver. The control packet overhead is the ratio from the total transmitted control packets to the total received packets. With the simulation parameters and the performance comparison metrics stated above, the simulation results are explained in the following four subsections.

We first investigated the impact of moving speed of the vehicles on the network performance. The average vehicle speed is varied from 5 to 30 m/s, the traffic flow is fixed at 0.1 veh/s. Figs. 4 and 5 show the packet delivery ratio of the overall traffic and the real-time traffic for the four schemes under different moving speeds, respectively. The two proposed mechanisms BAR-FLNNLC and BAR-BPNNLC perform better than the other two because these two schemes can effectively construct alternative routes to avoid link failure whenever link break or congestion warning is issued by the embedded neural networks. Notably, the packet delivery ratio for BAR-FLNNLC keeps fixed at around 95%. Meanwhile, these two figures also demonstrate the effectiveness of the fast learning neural networks. The performance of the traditional back-propagation neural networks degrade sharply due to significant dropping of the packets which is caused by the slow computation of neural networks during the training process. CVIA also shows a steep decrease owing to the significant increase of hop counts on the routing path when the moving speed of the vehicle increases to 15 m/s.

Fig. 6 shows the end-to-end delay of packets for the four schemes under different moving speeds. Notably, the end-to-end delay in this work includes the routing delay plus other processing
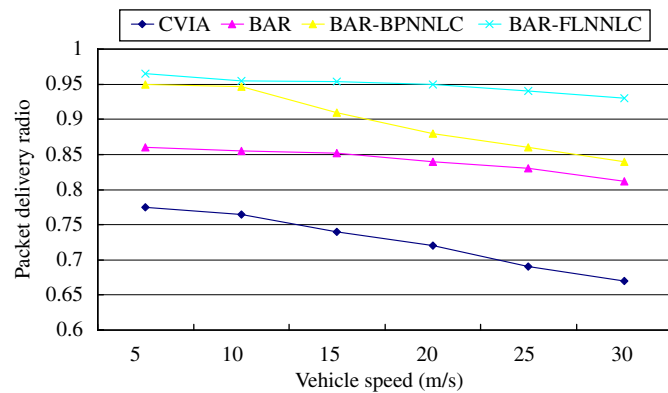

Fig. 6. End-to-end delay of the packets for the four schemes under different moving speeds.

Fig. 7. Control overhead for the four schemes under different moving speeds.

Fig. 4. Packet delivery ratios of overall traffic for CVIA, BAR, BAR-BPNNLC and BAR-FLNNLC under different moving speeds.

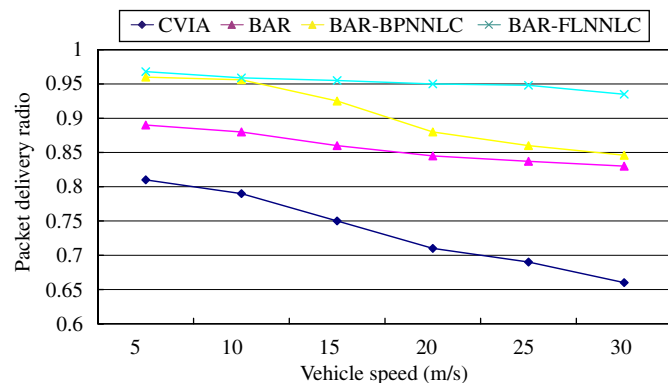**Fig. 5.** Packet delivery ratio of real-time traffic for the four schemes under different moving speeds.
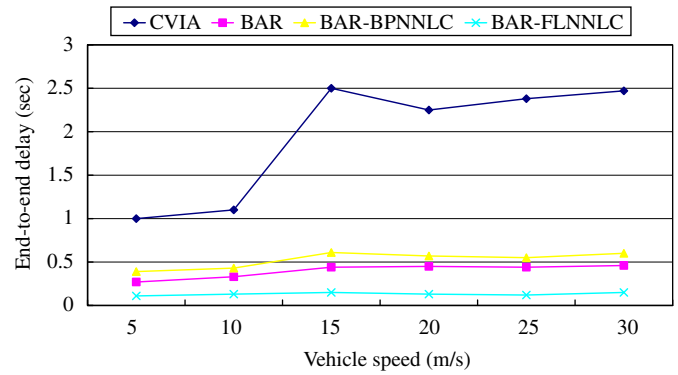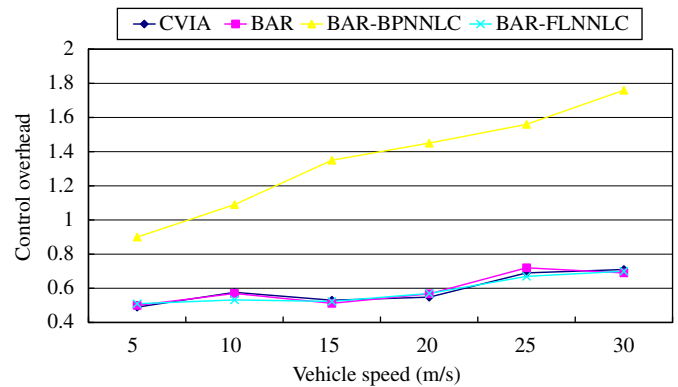
delays such as the computation of neural-network algorithm. The delay is measured for those packets from the mobile source vehicle to the mobile destination vehicle. The proposed BAR-FLNNLC scheme is effective in finding a shorter path by re-initiating route discovery process through the backup route mechanism. The reason behind this is that it not only transmits packets through shorter path but also prevents packet loss from link failure events. On the contrary, CVIA scheme performs the poorest and transmits packets through more hops than the other three schemes as well. The packets must go through more hops to reach the base station and this causes a longer transmission delay in CVIA scheme. Moreover, CVIA has to periodically construct transmission group at each time and the delay latency is thereby increased as well.

Fig. 7 shows the control overhead for the four schemes under different moving speeds. The control overhead is the required number of control packets that completes a packet transmission. Apparently, the control overhead for BAR-BPNNLC scheme ascends drastically when the moving speed of the vehicle is increased. It can be inferred that BAR-BPNNLC produces higher control overhead than the other three schemes since it needs to generate control messages to maintain the transmission group periodically. On the contrary, the control overhead for CVIA, BAR and BAR-FLNNLC schemes is slightly increased when the moving speed of the vehicle is increased. The proposed BAR-FLNNLC scheme maintains the routing path by means of piggybacking messages in the packets and hence the control overhead is lowered.

Figs. 8 and 9 demonstrate the impact of varied traffic flows on the packet delivery ratio in low speed. The mean traffic flow is set within the range of 0.1 and 0.5 veh/s. The average moving speed of each vehicle is set to 15 m/s because it is a very common driving

speed on local driveway. Meanwhile, varied traffic flows are generated from a Poisson distribution (Bose and Ioannou, 2003). The proposed BAR-FLNNLC scheme outperforms the other three schemes as expected. We believe it is because the link enhancement mechanism is able to assist the IVC in constructing the alternate route to transmit packet through congestion free route. On the other hand, CVIA and BAR schemes have to discard packets because of congestion and thus have poorer packet delivery ratio. Besides, the link enhancement mechanism in BAR-BPNNLC is unable to effectively boost the performance due to its expensive computation overhead.

Fig. 10 shows the end-to-end delay for the four schemes under different traffic flows on local roads. The proposed BAR-FLNNLC scheme performs much better than the other three schemes owing to its successful strategy on the deployment of alternative route used for link failure avoidance. The control overhead for the four schemes under different traffic flows on local roads is shown in Fig. 11. We can see that more control packets are required to keep network topology updated when the traffic flow becomes heavy in the schemes other than BAR-FLNNLC. The link failure prediction mechanism employed in FLNNLC keeps track of the topology change successfully, and
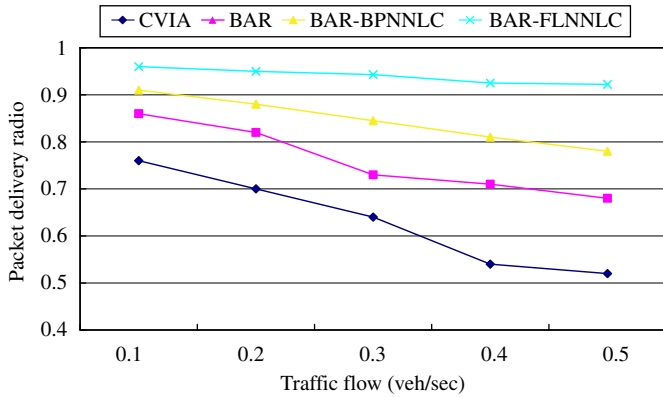

Fig. 8. Packet delivery ratio of overall traffic for the four schemes under varied traffic flows on local roads.
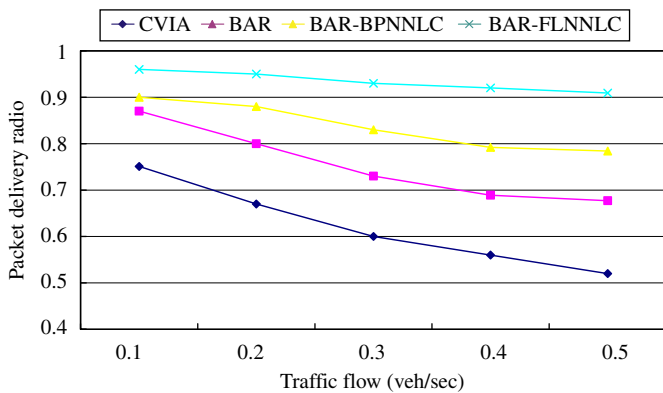

Fig. 9. Packet delivery ratio of real-time traffic for the four schemes under varied traffic flows on local roads.
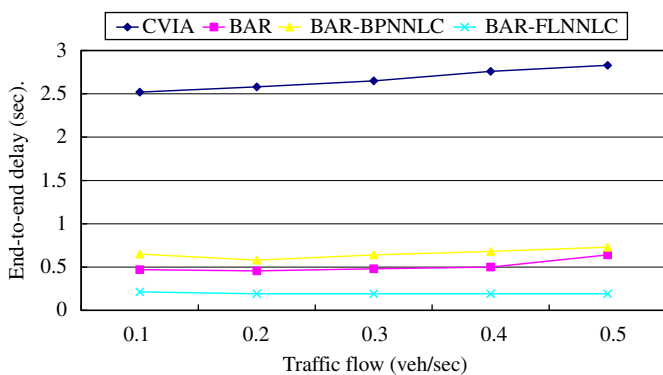

Fig. 10. End-to-end delay for the four schemes under different traffic flows on local roads.
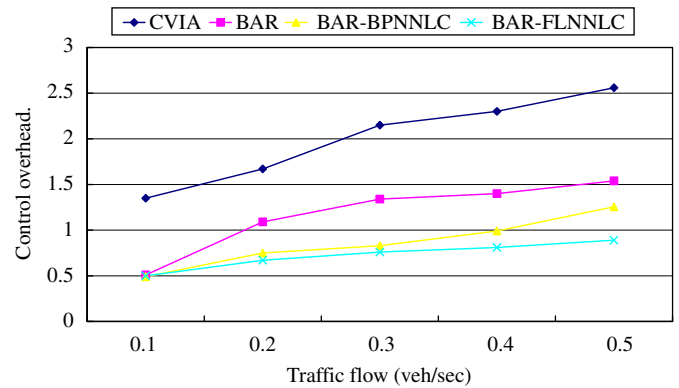

Fig. 11. Control overhead for the four schemes under different traffic flows on local roads.
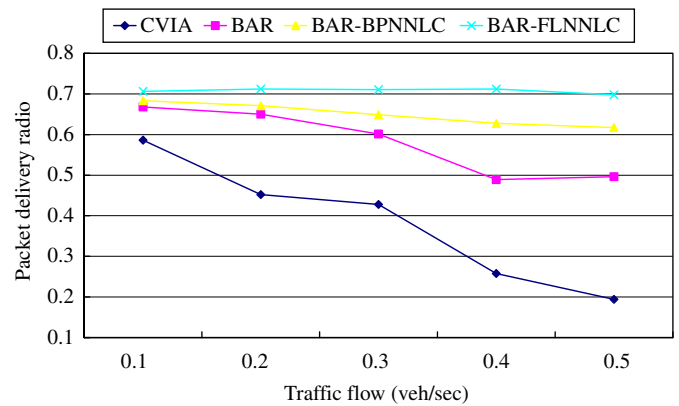

Fig. 12. Packet delivery ratio of overall traffic for the four schemes under different traffic flows in a high-speed moving environment.
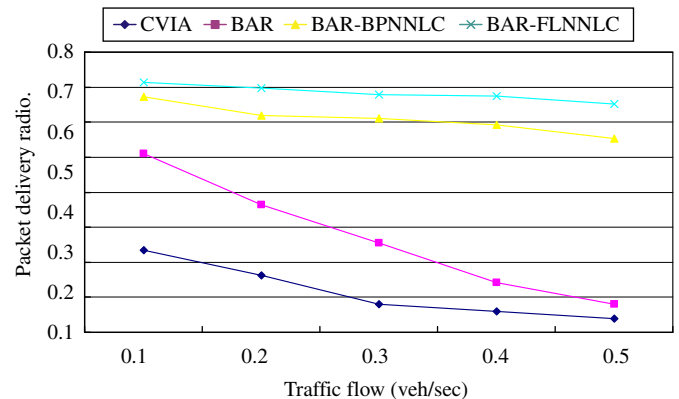

Fig. 13. Packet delivery ratio of real-time traffic for the four schemes under different traffic flows in a high-speed moving environment.
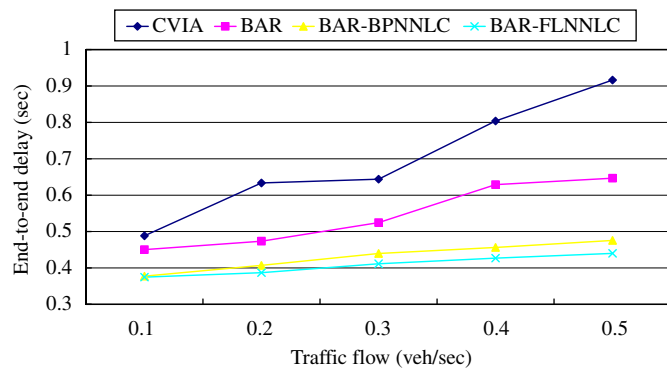
Fig. 14. End-to-end delay for the four schemes under different traffic flows in a high-speed moving environment.
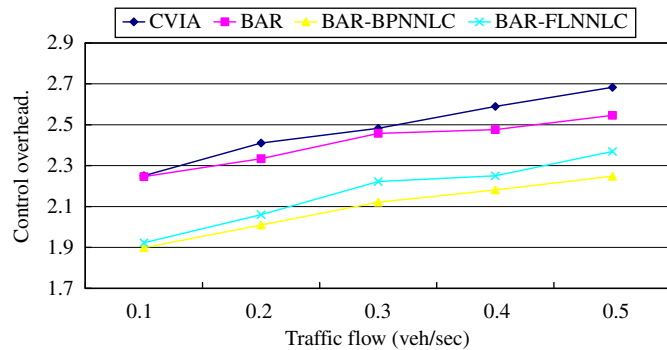

Fig. 15. Control overhead for the four schemes under different traffic flows in a high-speed moving environment.

reduces the need of the sending extra control packets in consequence.

Figs. 12 and 13 show the influence of different traffic flows on the packet delivery ratio on the highway. The average moving speed of each vehicle is set to 30 m/s. The four schemes in a high-speed moving environment perform much worse than in low speed due to quickly changed topology of the moving vehicle, which leads to unreliable connections among vehicles. Nevertheless, the proposed BAR-FLNNLC scheme significantly outperforms other schemes in a high-speed moving environment owing to the effectiveness of the congestion and link failure avoidance schemes. Notably, the packet delivery ratio for BAR-FLNNLC still keeps at around 70% even the number of vehicles is significantly increased in a high-speed moving environment. In contrast, the CVIA scheme has the worst packet delivery ratio for real-time traffic due to rapidly increased hop counts on the routing path in a high-speed moving environment.

Figs. 14 and 15 show the end-to-end delay for the four schemes under different traffic flows and the control overhead for the four schemes under different traffic flows in a high-speed moving environment, respectively. The BAR-FLNNLC scheme outperforms other three schemes as a result of successful management on the deployment of alternative route used for link failure avoidance in a high-speed moving environment. Moreover, the high-speed moving vehicles quickly change the topology of network, and this might result in the potential link breakage of the delivering routes. Extra control overhead packets are thus required to keep newly updated network topology information in the two schemes other than the proposed BAR-FLNNLC and BAR-BPNNLC schemes. CVIA scheme performs the worst due to a significant increase of hop counts on the routing path that is caused by the quickly changed topology of the moving vehicles.

Table 2
Comparison of train time and execution time for the BPNN and FLNN algorithms.

	BPNN	FLNN
Training time	4720 µs	21 µs
Execution time	80 µs	6 µs
Average accuracy rate	97.98%	90.67%

Table 3
t-test results of the paired "CVIA"–"BAR-FLNNLC" comparison on local roads.

Performance metric	Mean	StDev	SE Mean	T Value	P Value
Packet delivery ratio	−0.308	0.086	0.039	−7.977	0.001
End-to-end delay	2.473	0.134	0.060	41.331	0.000
Control overhead	1.280	0.343	0.154	8.336	0.001

Table 4
t-test results of the paired "CVIA"–"BAR-FLNNLC" comparison in a high-speed moving environment.

Performance metric	Mean	StDev	SE Mean	T Value	P Value
Packet delivery ratio	−0.324	0.155	0.070	−4.664	0.010
End-to-end delay	0.289	0.140	0.063	4.612	0.010
Control overhead	0.318	0.035	0.016	20.242	0.000

Table 2 gives the comparison of training time and execution time for the BPNN and the FLNN algorithms. The BPNN algorithm is much slower than the FLNN algorithm in terms of training time, and is not suited for the real-time applications, especially for the VANET environment where frequent training is required due to the ongoing network topology changes. In the 130 runs of the simulations, the average accuracy rate for BPNN and FLNN can reach up to 97.98% and 90.67%, respectively. Though the average accuracy for the BPNN is better than for the FLNN, the FLNN is more feasible than the BPNN due to its much shorter training time, which is an essential requirement in a volatile environment such as a VANET.

Tables 3 and 4 show the comparison of the *t*-test results of the three performance metrics for the CVIA and the proposed BAR-FLNNLC schemes on the local roads and the highway, respectively. The high-significance level as given by the results of *t*-test in Tables 3 and 4 once again reveals the effectiveness of the proposed BAR-FLNNLC schemes.

5. Conclusion

In this paper, QoS-aware routing mechanisms for VANETs are proposed to establish an effective routing path in IVC with assistance of roadside base stations. Alternate route construction and congestion avoidance mechanisms based on mobility patterns are presented to prevent the link failure caused by frequent change of network topology and the occurrence of congestion. A fast learning neural-network model is employed as the core modules in the link enhancement mechanisms to generate the link break and congestion indicators, which are piggybacked in the packets to inform the neighboring vehicles and roadside base stations. Meanwhile, a bandwidth consumption predictor embedded in each roadside base station is used to avoid dropping packets owing to inadequate bandwidth during handoffs. The simulation results showed that the proposed routing path construction and alternate route construction mechanisms can effectively prevent the link break caused by volatile vehicle

movements and traffic flows. The performance metrics including packet delivery ratio, control overhead, and end-to-end delay are significantly better than the representative IVC routing schemes in the literature. Meanwhile, the simulation results also support the effectiveness of using the fast learning neural network in the proposed work. In the future work, we will establish an autonomic policy-based management system and integrate it with the proposed routing mechanism to enhance the network resource management on safety applications and multimedia applications.

References

Bogenberger, R., Kellerer, W., Kosch, T., Reicher, T., Schwingenschlogl, C., Sties, P., Wagner, M., 2003. Virtual city portal—a multi-network personal information system for automobile users. In: IEEE/ITG International Workshop on Multi-radio Multimedia Communications.

Bose, A., Ioannou, P.A., 2003. Analysis of traffic flow with mixed manual and semiautomated vehicles. IEEE Transactions on Intelligent Transportation Systems 4 (4), 173–188.

Chen, S., Nahrstedt, K., 1999. Distributed quality-of-service in ad-hoc networks. IEEE J. Sel. Areas Commun. SAC-17 (8).

Cherry, S., 2003. Broadband: a Go-Go. IEEE Spectrum 40 (6), M. 20–M. 25.

Chung, Y., Cho, D.H., 2006. Enhanced soft-handoff scheme for real-time streaming services in intelligent transportation systems based on CDMA. IEEE Transactions on Intelligent Transportation Systems 7 (2), 147–155.

de Ridder, S. N., Elieff, C., Diesch, A., Gershenson, C., Pick Jr., H.L., 2002. Staying oriented while driving. In: Proceedings of the Human Factors and Ergonomics Society 46th Annual Meeting, Human Factors and Ergonomics Society, pp. 206–208.

DeLucia, P.R., Bleckley, M.K., Meyer, L.E., Bush, J.M., 2003. Judgments about collision in younger and older drivers. Transportation Research Part F, 63–80.

Elbahhar, F., Rivenq-Menhaj, A., Rouvaen, J.M., Heddebaut, M., 2001. Inter-vehicle communication based on ultra-wide band and CDMA techniques. In: Proceeding of the IEEE Intelligent Transportation Systems Conference, pp. 959–961.

Festag, A., Fussler, H., Hartenstein, H., Sarma, A., Schmitz, R., 2004. Fleetnet: bringing car-to-car communication into the real world. In: Proceedings of the 11th World Congress on ITS.

Füssler, H., Mauve, M., Hartenstein, H., Vollmer, D., Käsemann, M., 2002. A comparison of routing strategies for vehicular Ad-hoc networks. In: Proceedings of MOBICOM, Student Poster.

Huang, G.B., 2003. Learning capability and storage capability of two-hidden-layer feedforward networks. IEEE Transactions on Neural Networks 14 (2), 274–281.

Huang, G.B., Zhu, Q.Y., Siew, C.K., 2000. Extreme learning machine: theory and applications. Neurocomputing, 70, 489–501.

Huang, G.B., Zhu, Q.Y., Siew, C.K., 2006. Real-time learning capability of neural networks. IEEE Transactions on Neural Networks 17 (4).

Imielinski, T., Navas, J., 1996. GPS-based addressing and routing. Internet Engineering Task Force, Network Working Group, Request for Comments, RFC 2009, November 1996.

Kawamagari, Y., Fujii, M., Itami, M., Itoh, K., 2006. A study on DS/SS inter vehicle communication system using array antenna. In: Proceeding of the IEEE Intelligent Transportation Systems Conference, pp. 363–368.

Korkmaz, G., Ekici, E., Özgüner, F., 2006. A cross-layer multihop data delivery protocol with fairness guarantees for vehicular networks. IEEE Transactions on Vehicular Technology 55 (3), 865–875.

Linedemmeier, S., Boehm, K., Luy, J.F., 2003. A wireless data link for mobile applications. IEEE Microwave and Wireless Components Letters 13 (8), 326–328.

Maihofer, C., Eberhardt, R., 2004. Geocast in vehicular environments: caching and transmission range control for improved efficiency. In: 2004 IEEE Intelligent Vehicles Symposium, June 2004, pp. 951–956.

Mauve, M., Widmer, J., Hartenstein, H., 2001. A survey on position-based routing in mobile ad-hoc networks. IEEE Network Magazine.

Mesh Networks Inc., "Wirelessly Connecting the DOT'S: Mesh-enabled Solutions for Intelligent Transportation Systems," Available: ⟨http://www.meshnetworks.com⟩, 2000.

Morris, R., Janotti, J., Kaashoek, F., Li, J., Decouto, D., 2000. Carnet: a scalable Ad hoc wireless network system. In: Proceedings of the 9th ACM SIGOPS, pp. 61–65.

Namboodiri, Vinod, Gao, Lixin, 2007. Prediction based routing for vehicular Ad hoc networks. IEEE Transactions on Vehicular Technology 56 (4), 2332–2345.

Ni, S.Y., et al., 1999. The broadcast storm problem in a mobile ad hoc network. In: Proceeding of the ACM MOBICOM, pp. 151–162.

Obradovic, Z., Yan, P., 1990. Small depth polynomial size neural networks. Neural Computation 2, 402–404.

Sivak, M., Olson, P.L., Pastalan, L.A., 1981. Effect of driver's age on nighttime legibility of highway signs. Human Factors 23, 59–64.

Souryar, M.R., Vojcic, B.R., Pickholtz, R.L., 2006. Adaptive modulation in Ad hoc DS/CDMA packet radio networks. IEEE Transactions on Communication 54 (4), 714–724.

Sun, M., Feng, W., Lai, T., Yamada, K., Okada, H., Fujimura, K., 2000. Gps-based message broadcasting for inter-vehicle communication. In: Proceedings of ICPP, pp. 279–286.

Tokuda, K., Akiyama, M., Fuji, H., 2000. Dolphin for Intervehicle Communications System. In: Proceeding of the IEEE Intelligent Vehicles Symposium, pp. 504–509.

van Arem, B., van Driel, C.J.G., Visser, R., 2006. Impact of cooperative adaptive cruise control on traffic-flow characteristics. IEEE Transactions on Intelligent Transportation Systems 7 (4), 429–436.

Wu, H., Fujimoto, R., Guensler, R., Hunter, M., 2004. Mddv: a mobility-centric data dissemination algorithm for vehicular networks. In: Proceedings of the First ACM International Workshop in Vehicular Networks, pp. 47–56.

Xiang, C., Ding, S.Q., Lee, T.H., 2005. Geometrical interpretation and architecture selection of MLP. IEEE Transactions on Neural Networks 16 (1), 84–96.

Zhao, Jing, Zhang, Yang, Cao, Guohong, 2007. Data pouring and buffering on the road: a new data dissemination paradigm for vehicular ad hoc networks. IEEE Transactions on Vehicular Technology 56 (6).

Zhu, Q.Y., Huang, G.B., Siew, C.K., 2004. A fast constructive learning algorithm for single-hidden-layer neural networks. In: International Conference on Control, Automation, Robotics and Vision, pp. 1907–1911.