A Method for Sharing Traffic Jam Information using Inter-Vehicle Communication (Invited Paper)

Naoki Shibata, Takashi Terauchi[†], Tomoya Kitani[†], Keiichi Yasumoto[†], Minoru Ito[†], Teruo Higashino[‡]
Department of Information Processing and Management, Shiga University,
Hikone, Shiga 522–8522, Japan Email: shibata@biwako.shiga-u.ac.jp
[†] Graduate School of Information Science, Nara Institute of Science and Technology,
Ikoma, Nara 630–0192, Japan Email: {takash-t, t-kitani, yasumoto, ito}@is.naist.jp
[‡] Graduate School of Information Science and Technology, Osaka University,
Suita, Osaka 565-0871, Japan Email: higashino@ist.osaka-u.ac.jp

Abstract— **In this paper, we propose a method for cars to autonomously and cooperatively collect traffic jam statistics to estimate arrival time to destination for each car using inter-vehicle communication. In the method, the target geographical region is divided into areas, and each car measures time to pass through each area. Traffic information is collected by exchanging information between cars using inter-vehicle communication. In order to improve accuracy of estimation, we introduce several mechanisms to avoid same data to be repeatedly counted. Since wireless bandwidth usable for exchanging statistics information is limited, the proposed method includes a mechanism to categorize data, and send important data prior to other data. In order to evaluate effectiveness of the proposed method, we implemented the method on a traffic simulator NETSTREAM developed by Toyota Central R&D Labs, conducted some experiments and confirmed that the method achieves practical performance in sharing traffic jam information using inter-vehicle communication.**

I. Introduction

Recently, traffic jam is a huge social problem in urban area in many countries. Ministry of Land Infrastructure and Transport of Japan estimates that the economic loss caused by traffic jam in Japan every year is about 100 billion US dollars. This fact has brought a strong demand for drivers to know the congested areas and the estimated time required to get to their destinations. In recent years, several useful services are becoming available for drivers. ETC (Electronic Toll Collection) allows drivers to pass toll gate without stopping for payment and is installed in many cars. VICS (Vehicle Information and Communication System) [2] is a service using FM broadcast and optical beacons on the roadside to deliver traffic jam information to drivers so that their car navigation systems display congested areas/roads on the map and navigate them avoiding the congested areas. Although VICS is useful, there could be some time lag between the disseminated information and the real situation, since it collects all traffic jam information to one place (e.g., a central server), and disseminates it after processing. Also, VICS needs many devices installed on the roadside for monitoring traffic conditions, and thus it is costly to deploy VICS system to cover everywhere in a city.

In this paper, we propose a method to allow cars to autonomously collect and share traffic jam information using inter-vehicle communication based on IEEE 802.11, without using fixed infrastructure on the ground. This method allows drivers to estimate the time required to get to their destinations. The proposed method consists of (1) measurement of time to pass each route, (2) calculation of the statistics of time to pass each route by exchanging the measured time and statistics among cars, and (3) estimation of time required to get to destination. In the proposed method, the target road map is divided into fixed sub-regions called *area*s. Each car measures time to pass an area (called *area passage time*) for each entering/exiting pair of roads (called *linkpair*) of the area, and generates traffic information statistics from the information received from cars which passed the same pair of roads. By measuring average area passage time for each pair of roads crossing area boundary, difference between multiple routes with the same linkpair can be absorbed taking into account of waiting time at traffic lights and/or turning at intersections. This contributes to accurate estimation of time required for cars to get to their destinations.

When calculating traffic information statistics using broadcast-based inter-vehicle communication, the same data (area passage time and statistics) may be received and counted multiple times via neighboring cars, and this may lead to inaccurate results. In order to avoid this problem, we adopt a technique of the following two fold. First, we let each car attach its ID to the area passage time which it measured. Second, only when the number of area passage time data for each linkpair collected by each car reaches a predetermined number C, a statistics data is composed from these C data. The statistics data is attached a hash value calculated by C IDs of cars which produced the area passage time data. By these techniques, we can avoid large part of duplicated counting of same data. Since each area has multiple linkpairs and the number of statistics data increases as time passes, the data amount to be broadcasted by each car can be also too large in terms of bandwidth limitation of wireless communication (we suppose to use IEEE 802.11b). If each car sends packets too frequently, packet collision may frequently occur. To mitigate this problem, we prioritize packets based on the direction and location of the car which sends the packets and the locations of traffic information in the packets so that the traffic information is efficiently exchanged among cars.

In order to evaluate the usefulness of our method, we have implemented it on the traffic simulator called NETSTREAM [3]. As a result, we have confirmed that our method achieves practical information sharing ratio and estimates time to get to destination accurately enough.

II. Related Works

In Japan, VICS (Vehicle Information and Communication System)[2], which is a public service, provides latest traffic information to cars on the roads. VICS is a system which uses FM broadcast and various types of beacons to provide traffic information, and many car navigation systems use information from VICS to choose a route to the destination avoiding congested areas. VICS is a system which gathers traffic information to the information center, process the information and then broadcasts it toward cars. Thus, it has the following problems; (1) There is a little time lag between received information and current situation, (2) If all cars which have received information change route in the same way according to the information, the selected route will be congested quickly. (3) The devices for collecting traffic information are installed only on highways and trunk roads for cost-effectiveness, and thus information regarding to narrow roads may not be provided.

Pioneer Corp. has released ”Carrozzeria HDD Cyber Navi” which estimates traffic jams based on statistics of the pre-collected traffic information [1]. The Cyber Navi has the general road map data and traffic information of the past in its HDD. It can also get new road map data via a cell phone. The Cyber Navi estimates traffic jam based on the statistics and data from VICS, and selects a route taking into account of user’s preferences. It is capable of estimating traffic jam and time of arrival at the destination from statistics without VICS information, but it cannot handle accidental traffic jams caused by traffic accidents, road construction/repairing and so on.

Kanoh et al. have proposed a method to estimate traffic conditions from VICS information using clustering and cell automata based techniques[11].

Recently, realtime inter-vehicle communication is paid attention thanks to popularization of wireless LAN systems. By installing wireless LAN I/F on car navigation systems, cars can exchange traffic information with other cars using ad-hoc communication. For that purpose, several protocols have been proposed. Refs. [4], [5], [6], [7], [8] have proposed protocols for exchanging neighboring traffic information and situation using inter-vehicle communication. In [6], [7], a protocol for inter-vehicle communication with high information arrival rate is proposed. This protocol adjusts communication timings depending on traffic flows. The protocol is implemented on traffic simulator NETSTREAM[3]. In the research, a protocol has been designed to improve efficiency of information exchange taking into account of packet collision probability.

Korkmaz et al. have conducted some simulations of inter-vehicle and vehicle-to-road communication[8]. They use jamming signals called “black-burst” for congestion control.

These existing protocols are designed to propagate simple information to relatively close range, and it is difficult for these protocols to accurately estimate traffic jams of wide area and the arrival time to a destination.

As for prediction method of traffic jam, Abdulhai et al. have proposed a method using non-linear model[13], and Chrobok et al. have proposed a method which uses statistical information gathered in the past[12].

III. Proposed Method

The proposed method consists of a technique to gather statistical traffic information using inter-vehicle communication and a technique to estimate the time to get to the destination using the gathered information. In this section, first we describe the outline of the proposed method, and then explain the details of the techniques.

The objective of our method is to gather information using short range wireless communication, GPS and small computer on each car, without using fixed infrastructure on the ground.

In our method, we assume that each car has an onboard terminal with the following functionalities.

- IEEE 802.11 compliant wireless LAN device
- GPS receiver
- Hard disk drive to store traffic information
- Map data (on HDD)
- Computer with sufficient power for instantly processing received information

A. Overview

We assume that a given road map can be treated as a graph where each node and link correspond to an intersection and a road between intersections, respectively. The time to get to a destination from the current location of a car can be estimated theoretically by summing up time to pass each link to the destination. However, since there are a huge number of links on various routes to the destination, if we gather statistical traffic information for each link using inter-vehicle communication, the amount of data exchanged between cars may exceed available bandwidth of wireless communication. Also, in order to improve accuracy of the estimated time, we have to consider waiting time at each intersection caused by traffic lights, queue of cars turning left/right, and so on.

In the proposed method, the target geographical region is divided into square shaped *area*s with sides of several hundred meters length as shown in Fig. 1. We call the links through which a car enters and exits an area *incoming link* and *outgoing link*, respectively. A pair of incoming and outgoing links is called a *linkpair*. We collect time needed to pass each area for every linkpair. This time is called the *area passage time*. For example, in Fig.2, dotted lines indicate boundaries between areas. As for the area on the center of the figure, there are

5 links across boundary, indicated as $\alpha, \beta, \gamma, \delta$ and ϵ. When a car passes this area, the car passes two of these links, and thus there are $5 \times 4 = 20$ combinations of linkpairs. When a car crosses boundary of an area, the car records the current time. The area passage time is the difference of recorded time at incoming and outgoing links of the area.

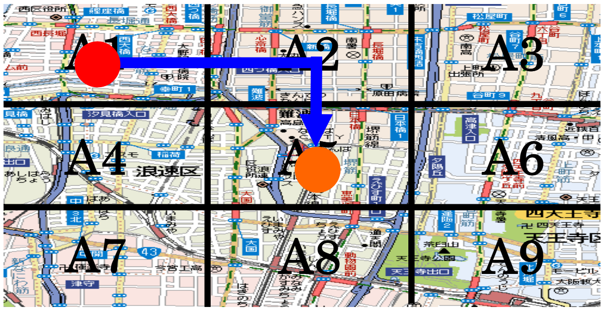

Fig. 1. Division into areas

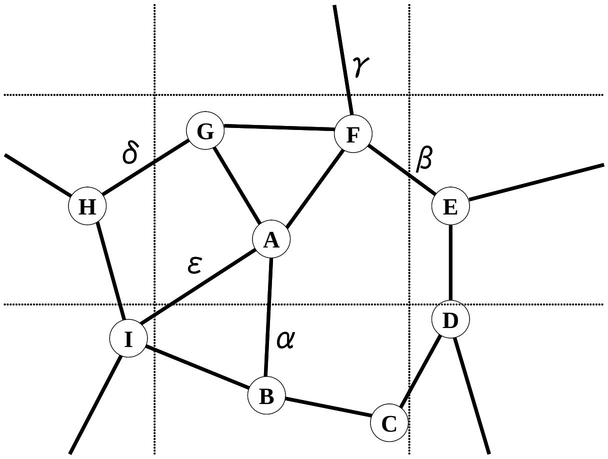

Fig. 2. Links across area boundaries

The traffic information (including area passage time) on each area has to be propagated to other cars which will pass the area in the future. So, we let cars to propagate the traffic information by repeated relays by cars. If we propagate the information to all directions equally, it takes long time for cars far from the target area to obtain its traffic information [6]. In the proposed method, we adopt a policy to use demand of cars at each area for other areas, and to propagate traffic information of each area to the areas with higher demand for the information, prior to the other areas.

B. Measuring area passage time

In order to accurately estimate arrival time, each car measures area passage time for each linkpair at each area. We assume that each area is assigned a unique ID number and that onboard terminal on each car has information regarding to locations of areas and their IDs.

Each car records area ID, link ID and the current time when entering an area. When the car leaves the area, time passed since the car entered the area is recorded as *area passage time*.

C. Avoiding redundant counting of area passage time

In order to calculate average area passage time for each linkpair, we have to collect data on area passage time for the same linkpair from multiple cars. If data generated by each car is simply exchanged with other cars through ad-hoc network, the data may be received by cars multiple times and treated as different data. It leads to an inaccurate calculation of area passage time. We introduce the following three mechanisms to avoid this problem.

Mechanism1: attaching a hash value to each area passage time: In the proposed method, each car retains data including a hash value generated by the car ID, the current time, and area passage time of several areas through which the car has passed recently. This data is called an *area passage record*. An area passage record consists of the following information.

$$(area\_id, il\_id, ol\_id, pass\_time, cross\_time, hash)$$

Here, $area\_id$ denotes the ID of the area, il_id and ol_id denote the link IDs of incoming and outgoing links to/from the area, respectively. $pass\_time$ denotes the time to pass through the area, and $cross\_time$ denotes the time when the car crossed the last area border. $hash$ denotes a hash value calculated from the car ID and the current time.

Each car periodically broadcasts area passage record toward neighboring cars. When a car receives an area passage record, it compares the car ID and the hash value of the record with those in its own records. If the hash values are equal, the received record is discarded.

As the number of area passage records retained in a car grows, the amount of data which the car broadcasts also grows. Due to limitation of available bandwidth in wireless network, we have to reduce the amount of data. In the proposed method, when the number of area passage records for a linkpair retained in a car reaches a predetermined threshold C (which is 3 to 5, typically), the values of these records are averaged by creating a statistics data called *area passage statistics*, and the original area passage records are removed.

The area passage statistics includes the following information (it can contain multiple records for different linkpairs).

$$(area\_id, il\_id, ol\_id, av\_pass\_time, num\_of\_cars, hash)$$

Here, $area\_id$ denotes the ID of the area, il_id and ol_id denote the link IDs of incoming and outgoing links to/from the area, respectively. av_pass_time denotes the average area passage time of $number\_of\_cars$ cars. $hash$ denotes a hash value calculated from the car IDs and the current time.

Mechanism2: attaching a hash value to each area passage statistics: Redundant calculation of area passage statistics also has to be prevented. So, when generating area passage statistics, a hash value is calculated from car IDs of original records and the current time, and attached to the area passage statistics data. When multiple area passage statistics data with the same linkpair are received by a car, it can discard redundant ones by comparing the hash values of the received data.

Each car periodically broadcasts the set of area passage statistics data which it retains. When a car receives a set of area passage statistics data from another car, it compares the hash value of each statistics data in the set with that in its own set, and adds it if it is not included in its own set.

Mechanism3: Removing redundancy between similar statistics data: Even with the above two mechanisms, area passage time of a car can be counted multiple times in the corresponding statistics. An example of this situation is that when car A and car B have the same sets of $C - 1$ area passage records, and A and B receive the different area passage records (e.g., from car X and car Y), respectively. Consequently, two different area passage statistics data with the same $C - 1$ records are generated. These statistics data may be merged into one statistics data in the future. In order to avoid this problem, we let each car checks for a specified time interval if the data received during the interval include area passage records or area passage statistics with the same hash value as those retained in the car. If so, the car discards all the data received during the interval.

D. Demand oriented propagation of statistics data

Gathering demand information: In the proposed method, demand information is gathered by exchanging and updating the *demand table* which consists of the amount of demand for each pair of two areas A and B, where cars at area A is going to pass area B in the future. We assume that the driver for each car inputs the destination into the car navigation system, and the system knows areas to pass through to get to the destination.

Each car periodically broadcasts its current position and expected areas which the car is going to pass. If a car receives this information, it merges the data with its table. The method to avoid redundant counting of the same data described above is used to prevent same data to be summed up multiple times.

The demand table has entries for each pair of two areas. The table retains the number of cars in each entry. For example, in Fig. 1, if car $C1$ running at area $A1$ is going to pass through areas $A2$ and $A5$, the car writes 1 in entries of columns $A2$ and $A5$ at line $A1$, and 0 in other entries as shown in Table I(a). If car $C2$ at area $A1$ sends car $C1$ a message which says car $C2$ is going to pass $A2$ and heading toward $A3$, car $C1$ updates entries of column $A2$ and $A3$ at line $A1$ and column $A3$ at line $A2$ as shown in Table I(b).

If we retain the table for large number of areas, the amount of information would be prohibitively large. We can avoid this problem by abstracting some information of the table so that entries of multiple remote areas are merged into one area.

Propagation of area passage statistics using demand information: Based on the demand table retained by each car, area passage statistics data are propagated to areas with higher demand prior to areas with lower demand. First, each car periodically broadcast area passage records, area passage statistics data and demand information. When a car receives the data, it adds the data into its own data if the car is going to pass an area with high demand for the received data. Otherwise, it just disposes the received data.

TABLE I
DEMAND TABLE

(a) Initial table entries

		Areas to be passed				
	AreaID	A1	A2	A3	A4	A5
@	A1	-	1	0	0	1
cur-	A2	0	-	0	0	1
rent	A3	0	0	-	0	0
pos.	A4	0	0	0	-	0
@	A5	0	0	0	0	-

(b) Table entries after update

		Areas to be passed				
	AreaID	A1	A2	A3	A4	A5
@	A1	-	2	1	0	1
cur-	A2	0	-	1	0	1
rent	A3	0	0	-	0	0
pos.	A4	0	0	0	-	0
@	A5	0	0	0	0	-

E. Data propagation and wireless bandwidth

Each car has to propagate information regarding to multiple areas, and it is impossible to send all information in one packet. In the proposed method, probability of sending packet including information regarding to an area is determined based on the congestion states and demand for the area. The more the area is congested or the more demand exists for information on the area, the data is sent at a higher probability. In order to realize this policy, we assign priorities among items of each car's retained data depending on their areas, as described below.

First, we suppose that a car is running at area $A5$ in Fig. 1. We divide each area into 9 subareas as shown in Fig. 3. We denote those subareas by $B1$ to $B9$. Information regarding to neighboring areas of area $A5$ are prioritized according to which subarea of $A5$ the car is running at. For example, if the car is in the central subarea ($B5$), we regard that areas $A2$, $A4$, $A6$ and $A8$ have highest priorities, $A5$ has medium priority, and $A1$, $A3$, $A7$, and $A9$ have the lowest priorities as shown in Fig. 4. In the figure, lower numbers show the stronger priorities.

IV. IMPLEMENTATION

In order to evaluate the usefulness of our method with realistic traffic flows on a realistic road system, we have implemented the proposed method on the traffic flow simulator called NETSTREAM [3].

NETSTREAM has been developed by Toyota Central R & D Labs. NETSTREAM was used to estimate traffic jam at Nagano Olympic Games in 1998 and generated a good estimation. NETSTREAM has a function to make more than

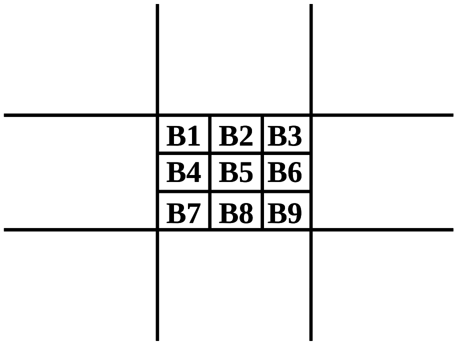

Fig. 3. Dividing an area into subareas

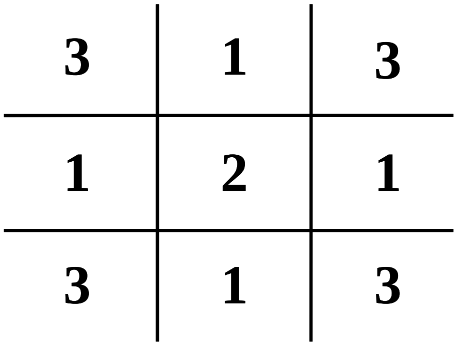

Fig. 4. Assigning priority based on subarea

1000 cars run on a given map simultaneously. It also has a function to construct an arbitrary road system consisting of roads (links) with legal speed limits and the number of lanes, intersections (nodes) with traffic lights specifying time intervals to change colors and so on using a graphical interface.

NETSTREAM simulates traffic flow on the given map as follows.

- read the map data
  - read the information on links and nodes
  - read other information such as the time intervals of traffic lights
- configure initial information of cars
  - decide the number of cars which follow each link or route
  - make each car run on the specified link within the legal speed limit
  - record logs including locations of all cars every second

A. Simulation of inter-vehicle communication

As a mean of inter-vehicle communication, we suppose to use IBSS standard of IEEE 802.11 and implemented it on NETSTREAM. We assume that communication range is 100m and that cars within this range can exchange packets. When multiple cars continuously broadcast packets at the same time, packet collision may occur. NETSTREAM has a function to record logs with locations of all cars every second, so it was easy to implement to packet collision detection at every second. However, in order to simulate inter-vehicle communication with fine granularity, location update every one second is too long. Thus, we divide one second into 100 time slots with 10ms lengths, and assume that each packet broadcast consumes only one timeslot. The size of each packet is assumed to be 1500 bytes. We assumed that packets are broadcasted in the timeslots decided at random. When a packet is broadcasted, the packet is registered in receive buffers of the corresponding timeslot of receivers in the radio range. If two or more packets are registered in a receive buffer for one timeslot, these packets are regarded as collided, and discarded.

Finally, for packets without collision, we simulate the probability of successful packet reception by the following formula.

$$P = \frac{-0.98x}{D} + 0.98 \tag{1}$$

Here, P is the probability of the successful reception, D is maximum distance to communicate (i.e., diameter of communication range), and x is the distance between two cars which try to exchange a packet. Note that we approximate attenuation of radio wave as the linear function of distance. As a result, if a packet does not collide with each other, the packet will be received by a car in the communication range at probability P defined above.

We have implemented the above mechanism using Visual C++ as a DLL of MS-Windows platform and installed it in NETSTREAM so that our DLL is called every second by NETSTREAM.

V. Experimental Validation

Based on our implementation explained in Sect. 4, we have conducted simulation to investigate how traffic information is generated and propagated among cars using inter-vehicle communication.

A. Simulation Configuration

We used the following configuration in our experimental simulation.

- All cars are equipped with a system which carries out generation and broadcast of area passage records as well as reception, update and re-broadcast of area passage statistics data for each linkpair through an IEEE 802.11b wireless LAN device.
- Direct communication range (radio range) of each car is 100m.
- Each car in the communication range of a car which has broadcast a packet receives the packet at the probability defined in expression (1) as long as the packet collision does not occur.
- As explained in Sect. 4, since each second is divided to 100 timeslots, packet transmission and reception between cars is simulated per timeslot (10ms).
- Simulation time is up to 60 minutes (i.e., 360,000 timeslots).
- The size of each packet is 1500 bytes and each car can send at most one packet for each timeslot.

- The legal speed limit of each road is 60Km/hour.
- As threshold C for making area passage statistics data, we used $C = 5$.
- The map is divided to squares with 300m sides.
- Area passage statistics data is discarded after 30 minutes from its creation time.

We used a general road map consisting of two main roads crossing at the center and several byroads as shown in Fig. 5. The map has 1.2Km sides, and 29 nodes (intersections) and 78 links (roads) in it. In the simulation, we gradually increased the number of cars running on the map from 0 to around 300 for the maximum so that the density of cars in each area is sufficient for inter-vehicle communication.

We let each car broadcast area passage data five times in 5 seconds right after it is generated as well as each area passage statistics data which the car retains, three times every 10 seconds. The packet is broadcasted in the timeslot decided at random.

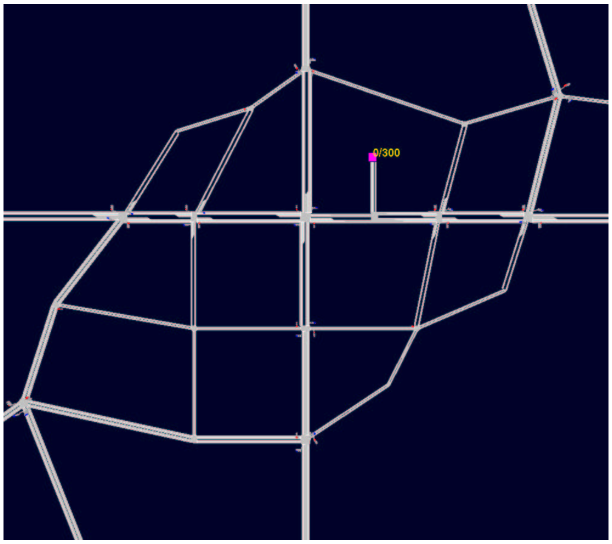

Fig. 5. Road map used for simulation

B. Experimental Results

First we investigated the ratio of the successful packet transmission/reception. According to our experiments, among all packets which were broadcasted, 12.1 % of them collided and 39.2 % were not received by any car due to the probability defined by expression (1). Thus, the remaining 48.7 % of packets could be received by cars. Here, if a packet is broadcasted by a car and two cars received the same packet, we regard that the two packets were broadcasted and received.

Next, we conducted the following four experiments, in order to evaluate our proposed method in terms of information sharing ratio, accuracy of estimated time to get to destination, and effectiveness of our techniques: prioritized transmission of data and redundancy avoidance. In these experiments, we executed simulation for 30 to 60 minutes and calculated the average time to pass each linkpair for cars which actually passed the linkpair. We call this time as *actual passage time* representing the ideal case. We selected 9 linkpairs (referred to as A, B, C, D, E, F, G, H and I) in the map of Fig. 5 for evaluation.

TABLE II

AVERAGE AREA PASSAGE TIME OBTAINED BY OUR METHOD

linkpair	average area passage time by our method (sec)	actual passage time (sec)	difference (%)
A	13.1	13.0	0.7
B	38.9	35.5	9.6
C	24.6	36.2	32.0
D	48.8	49.4	1.2
E	67.6	58.0	16.6
F	68.2	74.2	8.1
G	50.0	80.4	37.8
H	67.6	92.8	27.0
I	88.1	93.8	6.1

TABLE III

ESTIMATED TIME TO GET TO DESTINATION BY OUR METHOD

linkpairs	estimated time by our method (sec)	actual passage time (sec)	difference (%)
a	22.0	28.0	21.4
b	62.3	56.3	10.7
c	62.6	62.6	0.0
d	76.3	90.0	15.2
e	91.1	95.2	4.3
f	98.7	95.5	3.4
g	124.1	128.0	3.9

Experiment1: Accuracy of time to pass areas which cars obtained by our method: In this experiment, at the end of simulation (i.e., 60 minutes point), we picked up cars on the map which retained area passage data and/or area passage statistics data, and calculated the average value among them. We also compared the value with the actual passage time. Here, smaller difference is better. We show the result in Table II.

According to Table II, we see that for 5 of 9 linkpairs, the differences were less than 10%, which we think our method is accurate enough for practical use. For linkpair G, there is a big difference (37.8%). However, the actual passage time for the last 10 minutes of simulation was around 59.0 sec and this is not far from the average area passage time (50.0 sec) obtained by our method.

Experiment2: Estimation of time to get to destination over areas: In order to estimate time to get to destination, for two linkpairs which pass through two neighboring areas, we picked up cars which retained data on the two linkpairs on the map and calculated the average of sum of area passage time for the two linkpairs. We also compared the value with the actual passage time for the two linkpairs. The result is shown in Table III. Here, we used 7 combinations of linkpairs: a, b, c, d, e, f and g.

According to Table III, when the time to get to destination is small, the difference is likely large. However, when the time is more than 90 second, the differences were less than 5 %. For the cases that actual passage time varied as time passed,

TABLE IV

EFFECTIVENESS OF PRIORITIZED DATA TRANSMISSION

linkpair	without prioritization (sec)	with prioritization (sec)	actual passage time (sec)
A	0	0	13
B	0	34	35.5
C	23	29.7	36.2
D	45.5	47.2	49.4
E	37.1	52.1	58
F	62.1	68.8	74.2
G	64.3	56.7	80.4
H	83.4	85.1	92.8
I	76	88.5	93.8

TABLE V

EFFECTIVENESS OF REDUNDANCY AVOIDANCE

linkpair	without redundancy avoidance (sec)	with redundancy avoidance (sec)	actual passage time (sec)
A	0	0	13
B	42	34	35.5
C	24.1	29.7	36.2
D	52.3	47.2	49.4
E	69.1	52.1	58
F	67.1	68.8	74.2
G	69.1	56.7	80.4
H	84.1	85.1	92.8
I	117.4	88.5	93.9

the difference became large.

Experiment3: Effectiveness of prioritized data transmission: In order to investigate the effectiveness of the prioritized data transmission explained in Sect. III-E, we conducted simulation for both cases with and without this mechanism and compared the average area passage time by our method with the actual passage time. The result is shown in Table IV.

Table IV suggests us that when this mechanism is used, the estimated time becomes close to the actual time. Area passage statistics data for linkpairs A and B were completely lost when we did not use the mechanism. For A, even with prioritization, the data was lost since only two cars actually passed this linkpair in the simulation.

Experiment4: Effectiveness of redundancy avoidance mechanism: We conducted simulation using a mechanism for avoiding redundant count of the same area passage time to statistics explained in Sect. III-C. The results are shown in Table. V.

As we see in Table V, when we use the redundancy avoidance mechanism, the average area passage time by our method became quite close to the actual passage time. On the other hand, when we did not use the mechanism, the area passage time was likely close to the mode of actual passage time. This tendency was seen in all roads in the map. This means that our redundancy avoidance mechanism improves the accuracy of the area passage time to a certain extent.

VI. CONCLUSION

In this paper, we proposed a method for generating and sharing traffic information on roads using inter-vehicle communication based on IEEE 802.11b. We implemented the proposed method on the traffic simulator NETSTREAM, and through simulation with realistic traffic on a typical road system, we confirmed that our method achieves the traffic information sharing among cars at practical level.

At present, our proposed method allows drivers to obtain traffic information in rather neighboring area. We need to expand the information propagation range and to allow drivers to estimate time required to get to relatively far destinations. Also, in order to update and keep accurate traffic information statistics continuously, sufficient number of cars must exist in each area. In [10], we already proposed a technique to mitigate the latter problem using message ferrying technique [9]. For future work, we will challenge the above problems to make our method more practicable.

ACKNOWLEDGMENT

We would like to thank to Mr. Eiji Teramoto, Mr. Hironobu Kitaoka and Ms. Hiroko Mori of Toyota Central R&D Labs., Ltd. for their support to implement our system in NETSTREAM.

REFERENCES

[1] Pioneer : HDD Cyber Navi, http://pioneer.jp/carrozzeria/cybernavi/

[2] Ministry of Land Infrastructure and Transport : “The system outline of VICS,” http://www.its.go.jp/etcvics/vics/

[3] E. Teramoto, M. Baba, H. Mori, H. Kitaoka, I. Tanahashi, Y. Nishimura, et.al.: “Prediction of Traffic Conditions for the Nagano Olympic Winter Games Using Traffic Simulator : NETSTREAM,” Proc. of 5th World Congress on Intelligent Transport Systems, Vol.4, pp.1801–1806, 1998.

[4] B. Xu and O. Wolfson : “Opportunistic Resource Exchange in Inter-vehicle Ad-hoc Networks,” 2004 IEEE International Conference on Mobile Data Management (MDM 2004), 2004.

[5] T. Nadeem, S. Dashtinezhad and C. Liao : “Traffic view : A Scalable Traffic Monitoring System,” 2004 IEEE International Conference on Mobile Data Management (MDM 2004), 2004.

[6] M. Saito, M. Funai, T. Umedu and T. Higashino : “INTER-VEHICLE AD-HOC COMMUNICATION PROTOCOL FOR ACQUIRING LOCAL TRAFFIC INFORMATION,” Proceedings of the 11th World Congress on ITS, CD-ROM 4066.pdf , 2004.

[7] M. Saito, J. Tsukamoto, T. Umedu and T. Higashino : “Evaluation of Inter-Vehicle Ad-hoc Communication Protocol,” Proceedings of the 19th International Conference on Advanced Information Networking and Applications (AINA2005), pp. 78–83 , 2005.

[8] G. Korkmaz, E. Ekici, F. Ozguner and U. Ozguner : “Urban Multi-Hop Broadcast Protocols for Inter-Vehicle Communication Systems,” Proceedings of ACM Workshop on Vehicular Ad Hoc Networks (VANET 2004), pp. 76–85, 2004.

[9] W. Zhao, and M. H. Ammar : “Message Ferrying: Proactive Routing in Highly-partitioned Wireless Ad Hoc Networks,” Proc. of 9th IEEE Workshop on Future Trends of Distributed Computing Systems (FTDCS 2003), pp.3008–3014, 2003.

[10] T. Shinkawa, T. Terauchi, T. Kitani, N. Shibata, K. Yasumoto, M. Ito and T. Higashino : “A Technique for Information Sharing using Inter-Vehicle Communication with Message Ferrying,” Proc. of 2006 International Workshop on Future Mobile and Ubiquitous Information Technologies(FMUIT 2006), pp.221–225, 2006.

[11] H. Kanoh, T. Furukawa, S. Tsukahara, K. Hara, H. Nishi and H. Kurokawa : “Short-Term Traffic Prediction Using Fuzzy C-Means and Cellular,” IEEE International Conference on Intelligent Transportation Systems (ITSC 2005), pp.984-988, 2005.

[12] R. Chrobok, O. Kaumann, J. Wahle and M. Schreckenberg : “Different method of traffic forecast based on real data,” European Journal of Operational Research, Vol.155, pp.558-568, 2004.

[13] B. Abdulhai, H. Porwal and W. Recker : “Short-term traffic flow prediction using Neuro-Genetic algorithms,” ITS Journal, Vol. 7, no. 1, pp. 3–41, 2002.