Research Article

A Geo-Aware Taxi Carrying Management System by Using Location Based Services and Zone Queuing Techniques on Internet of Things

Chyi-Ren Dow, Duc-Binh Nguyen, Shr-Chen Wang, Shiow-Fen Hwang, and Ming Fong Tsai

Department of Information Engineering and Computer Science, Feng Chia University, No. 100, Wenhua Road, Seatwen, Taichung 40724, Taiwan

Correspondence should be addressed to Chyi-Ren Dow; crdow@fcu.edu.tw

Received 29 September 2015; Accepted 16 November 2015

Academic Editor: Jong-Hyouk Lee

Copyright © 2016 Chyi-Ren Dow et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Taxi plays a crucial role in the transportation system because of the characteristic that can be hailed conveniently. Most of the taxi drivers obtain passengers by hunting on the road or waiting in a fixed taxi queuing point; however these methods have poor performance, high vacancy rate, and several critical problems such as air pollution and foul up traffic. This study proposed a taxi carrying management system by using location based services and zone queuing techniques on Internet of things. The proposed system allows drivers to both hunt on the road and wait in a queuing zone. A queuing table is used in the control center and neighbor tables are used in RSUs for zone queuing establishment. Joining and leaving mechanisms are developed for zone queuing management. To enhance service efficiency and quality, we present a scheme to prevent the ping-pong effect which is based on the location based services, a hunting rate calculation scheme, and a path planning service for taxi drivers according to the history carrying record. PRISM is used to simulate the proposed system, and the results indicated that our scheme outperforms the waiting and hunting models in terms of number of customers, vacancy rate, and profit.

1. Introduction

Internet of vehicles (IoV) is an emerging research problem in recent years. It is a complex integrated network system which converges the mobile Internet and the Internet of Things (IoT) by comprising of various vehicles. It is a converged technology that encompasses information communication, environmental protection, energy conservation, and safety. Numerous applications use vehicle-to-vehicle [1], vehicle-to-sensor [2], and vehicle-to-infrastructure [3] communications in IoV, such as traffic event alarms, advertisement broadcasting, and entertainment services. Because of the characteristics of IoV networks, vehicles can conveniently transmit their traveling information and receive information from other vehicles with supported hardware and software.

Taxis play a crucial role for the convenience of traffic transportation, which reflects the levels of economic development and civilization of a city. Internet of Taxis is an important part of a smart city. The characteristics of IoV

help taxi drivers to easily hail customers and reduce the vacancy rate. Presently, there are three common taxi carrying models. The first model is hunting, and in this model, a taxi driver finds passengers on the street through luck and experience. This consumes considerable time and gasoline. The second model is fixed queue in which taxis wait and queue for passengers at specific places. Most of these places are around city hot spots such as a train station with high demand for taxi services. In this model, taxi drivers can pick up a passenger without consuming considerable gasoline. However, the waiting time increases when there are a high number of taxis in the queue. The third model is dispatching, and this model comprises drivers who join a fleet, such as “Taiwan Taxi” or “M-Taxi.” A passenger can call the fleet for a taxi by phone or through the Internet, and the fleet dispatches a taxi to a specific place. This increases the opportunity for drivers to obtain passengers. However, the drivers must pay for each call of dispatching, even if the passenger does not show at the specified place.

Each of these traditional taxi carrying models has its own pros and cons. The major disadvantages of the waiting model are long waiting time and blocked traffic when the number of taxis in the queue is higher than the capacity of the waiting area. It is vital to demarcate a fixed point of the waiting model to a larger area for reducing the waiting time and blocked traffic. However, ping-pong effect [4] is a potentially undesirable phenomenon, in which the system needs to perform handovers frequently between the same pair of demarcated areas back and forth, when drivers roam around the boundary of the areas in a short time period. Ping-pong effect can cause inefficiency, dropping from the queue of queuing zone, and degrading of the system performance. Coverage parameters, vehicle's location area and its movement, are the main considerations that can cause the ping-pong effect. Furthermore, the hunting model is more favorable than the waiting model in terms of average number of customers and vacancy rate because drivers actively hunt for customers on the street. The waiting model is more favorable than the hunting model in terms of gas consumption that directly affects a taxi driver's profit. In order to obtain advantages of both the hunting and waiting models, a hybrid model should be investigated and developed to allow drivers to both hunt on the road and wait in a queue.

To resolve the aforementioned problems, this study proposed and developed a geo-aware taxi carrying management system by using location based service (LBS) and zone queuing techniques on Internet of vehicles. This is a distributed system, and in this system, the location based services were used in zone queuing establishment and service areas are demarcated based on a geogrid [5]. Furthermore, in the proposed system, each service area can function independently as a queuing zone. This study also implemented a system prototype with wireless access in vehicular environments/dedicated short-range communications (WAVE/DSRC) to provide communications and user interfaces for both the control center and vehicles to verify the feasibility of the proposed system. In the experimental results section, we simulated and proved that the efficiency of our proposed scheme is better than the waiting and hunting methods via PRISM [6] which is a module checker supporting many probabilistic models. PRISM uses a simple and state-based language to express probabilistic models and automatically analyze data patterns. It also includes a discrete-event simulation engine, providing support for approximate/statistical model checking, and implementations of various analysis techniques, such as quantitative abstraction refinement and symmetry reduction.

The rest of this paper is organized as follows. Related work is reviewed in Section 2. In Section 3, the zone queuing scheme is summarized. Section 4 presents the prototype of the implemented user interface and communication setting. Section 5 presents the experimental results, and Section 6 presents the conclusion and future work.

2. Related Work

This section presents some related work, including geo-aware service's research for taxi and queuing techniques.

The geo-aware service is well-developed presently and uses the GPS to observe and collect information. Several studies used sensors and GPS to monitor environments in specific places for preventing and promptly notifying natural disaster [7], to propose more convenient traveling methods in transportation [8]. Because of the fast growing of intelligent transportation systems (ITS), numerous applications were proposed such as traffic control [9], road safety [10], and routing path planning [11, 12]. A traffic information system is proposed [13] which used RSUs to collect vehicle information in each road segment and provided traffic flow state to drivers for safe and shortest distance to destination. Tornell et al. [14] proposed an application for smart phones based on the enhanced Message Dissemination based on Roadmaps protocol (eMDR). The application integrates a navigation system and wireless communication device and uses ITS information to avoid critical-mission collision and achieve safety driving. IoV also is considered as ITS combined with Vehicular Ad Hoc Network (VANet) techniques to provide communications between vehicles and infrastructure by using road side units (RSUs) and on-board units (OBUs) via Internet environment [15].

There are three common research problems regarding taxi. The first problem involves real-taxi environment modeling. Most of researches used an amount of real-taxi GPS traced records to analyze statistics and emulate the taxi module, such as behavior of taxi drivers [16] and passenger generators [17]. The next problem involves designing a protocol to achieve the goal of curtailing waiting time or saving gas. Chen et al. [18] proposed a dynamic taxi-sharing protocol which utilized traffic information and ITS technology to avoid traffic congestion. This work, though getting fuel-saving and pollution reducing when the number of passengers willing to share taxis increases dramatically, is still not well enough to reach high business efficiency. Sheu et al. [19] proposed a distributed taxi hailing protocol which aims to hail a taxi with shortest distance under the rules of road signature and reduce vacancy rate as much as possible. However, this method planned and recommended path without the highest possibility to hail customers which may help to increase income for a taxi driver.

The final problem involves implementing application systems for taxi environments. Hosni et al. [20] developed a shared taxi service system which benefits both the taxi drivers and the passengers. In these studies, a passenger can make a reservation on the Internet or call the service center to book a taxi. The communication system between taxi and passenger has been implemented by Liu et al. [21] in which passengers can join an occupied taxi on road if they have the same destination. The system also provides trip history model to analyze behavior of passengers in a specific place, and this model can reduce vacancy rate of taxis.

Regarding scheduling schemes, Yuan et al. [22] proposed a method to guarantee strict fairness and utilize prediction better in parallel job scheduling. McKeown [23] showed scheduling algorithm called iSLIP. An iterative, round-robin algorithm, iSLIP, can achieve 100% throughput for uniform traffic, simple to implement in hardware and extensively used in various applications. Gabale et al. [24] classify scheduling

algorithms with problem setting, problem goal, type of inputs, and solution techniques in wireless mesh networks. Fan and Quan [25] proposed harmonic-aware multicore scheduling based on the rate monotonic scheduling (RMS) policy to guarantee the schedulability of real-time tasks. A dynamic scheduling algorithm proposed by Ren and van der Schaar [26] can be used in wireless cloud computing.

3. Zone Queuing Scheme

This section presents the proposed zone queuing scheme, including zone queuing establishment, zone queuing management, and path planning protocol.

3.1. Zone Queuing Establishment. The first part of the proposed zone queuing scheme involves zone queuing establishment, which can be divided into area demarcation, center selection, and weight calculation formula. The area demarcation depends on the number of passengers in the whole area; this is because most of the existing queuing zones are located in high service demand areas, such as Taipei city. The center is the brain of the queuing zone and controls all the actions of the queuing zone and provides planning path service for drivers. The weight calculation formula is used to calculate the weight for each grid, and paths are planned based on this weight.

3.1.1. Area Demarcation and Center Selection. For the area demarcating, a queuing zone is established in specific places, such as department stores, hospitals, stations, night markets, or other hot spots that have a high service demand for taxis. Thus, the first step in establishing the queue involves defining the size of a queuing zone based on the service requirement. The boundary was determined using GPS technology and this is important to deal with the ping-pong effect (see Section 3.2 for more details). In order to obtain queuing information

from the control center, this study assumed that each taxi is equipped with wireless communication devices. The taxi also requires a GPS sensor and a digital map to determine its location.

To ensure the communication between a taxi and the center, the second step, this study used geogrids that cut the queuing area into several grids; the grid size was based on the transmission range R of the devices. In each grid, a RSU was developed as the grid center. Normally, each RSU can communicate with the service center and other RSUs through wired networks. This model is used to collect information from taxis and serves as a hailing stand for passengers who can use wireless communications, such as Wi-Fi, 3G, or 4G, to hail a taxi.

The third step involves determining the control center of the queuing zone. The control center plays a vital role in this scheme because it manages queuing processes, such as joining the queue, leaving the queue, and calculating the average wait time. To ensure efficient data collection, the node located at the center of the queuing zone was used as the control center because the distance from this node to each of the other nodes is slightly similar.

3.1.2. Queuing Table in the Control Center. The control center manages all processes in a queuing zone. Thus, it must maintain a queuing table to manage such processes. Equation (1) shows the queuing table. Queuing number is a serial number, and lower value means higher priority. Vehicular ID is a unique plate number. RSU number records the RSU that enables a taxi to transmit messages. Timestamp is used to determine which messages must be updated or deleted. Because only the newest message is recorded in the queuing table, the control center can communicate with the taxi via the RSU number.

Queuing Table in the Control Center

Queuing Number	Vehicular ID	RSU Number	Timestamp	TTL
----------------	--------------	------------	-----------	-----

(1)

Neighbor Table in RSU

Vehicular ID	State	Timestamp	TTL
--------------	-------	-----------	-----

(2)

Basically, a taxi essentially sends “hello” messages periodically to the RSU in its grid for updating its location if it is in the queue. When the RSU receives a hello message, it compares message content with information in the neighboring table. If the taxi information is not in the table, the RSU transmits an updated message to the control center and updates its neighbor table. Otherwise, it updates only its neighboring table. In addition to periodically updating the state of queuing, the RSU sends an advertisement message when it receives hailing requests from passengers. Each taxi can send a response when it receives the message, even if it is not in the queue. Equation (2) illustrates the neighbor table

format for the RSU; in this table, state denotes whether a taxi is in the queue. Ping-pong time to live (TTL) denotes the time used for the taxi to prevent the ping-pong effect.

3.2. Zone Queuing Management. The second part of the proposed scheme is zone queuing management, which involves joining and leaving mechanisms and provides a scheme for preventing the ping-pong effect. In the joining and leaving mechanisms, the workflow for a taxi to join or leave the queue is presented in detail. The ping-pong effect is a prevalent problem in wireless communications and is discussed in Section 3.2.2.

3.2.1. Joining and Leaving Mechanisms. When a taxi enters a queuing zone, it transmits a joining message to the control center via the RSU in its grid, and the control center returns

FIGURE 1: Workflow of joining or leaving the queue.

the queuing number and planning path to the taxi after receiving a joining message. The taxi joins a queue successfully when it receives the response message, and it starts transmitting update message periodically. Figure 1 shows the workflow for joining or leaving a queue.

When a taxi driver intends to obtain passengers by hunting or to leave the queue for personal reasons, the driver can transmit a leaving message by using the console queuing function. When the control center receives a leaving message, it deletes the information of the taxi from the queuing table and responds with a message confirming the completion of the leaving process. If a taxi leaves without transmitting a leaving message, the control center deletes it from the queuing table when the timestamp is not updated in a specific period.

Because the taxi periodically transmits updating messages to the RSU after successfully joining the queue, the control center can accurately locate the taxi according to the RSU number. When the control center receives a leaving message from a taxi, it examines the RSU number to ensure whether the message is legal. If someone attempts to transmit a fake leaving message, the control center ignores the message if its RSU number does not match.

3.2.2. Prevention of the Ping-Pong Effect. The ping-pong effect is a prevalent problem in cellular network systems. It occurs when users are located on the boundary of two base stations, and it requires frequent handover to keep service in operation, thus leading to a long delay time and unfavorable performance. In the proposed system, this problem is encountered when a taxi is roaming around the boundary of a queuing zone. If the boundary problem is not handled, the taxi is deleted from the queue if the control center cannot locate it, and the taxi must rejoin the queue and wait for a longer period. To ensure connection between a taxi and the control center, the taxi must determine whether it is prepared to leave the queue; the first step involves defining a boundary area.

Therefore, the boundary area comprises two domains: warning domain and switching domain.

The formulas of these two domains are defined as follows.

Warning domain: taxi in the grid of boundary RSU and its location (x, y)

$$\left[(x, y) \mid \left\{ \begin{array}{l} x_{\min} + \alpha < x \cup x \leq x_{\max} - \alpha \\ y_{\min} + \alpha < y \cup y \leq y_{\max} - \alpha \end{array} \right\} \right]. \quad (3)$$

Switching domain:

$$\left[(x, y) \mid \left\{ \begin{array}{l} x_{\min} < x \leq x_{\min} + \alpha \cup x_{\max} - \alpha \leq x < x_{b\max} \\ y_{\min} < y \leq y_{\min} + \alpha \cup y_{\max} - \alpha \leq y < y_{b\max} \end{array} \right\} \right]. \quad (4)$$

$x_{\min}$ is the minimum and $x_{\max}$ is the maximum longitude, $y_{\min}$ is the minimum and $y_{\max}$ is the maximum latitude of queuing zone, and α is a threshold for flexible boundary area. In this study, we suggest α in a range from 1/3 to 2/3 of d , whereas d is an offset between $x_{\min}$ and longitude value of nearest boundary RSU.

If an in-queue taxi is in the warning domain of the boundary area, the driver is warned about leaving the queuing zone via a graphical user interface. The warning is continually shown until the taxi is out of the boundary area. If the taxi continues moving to the switching domain from the warning domain, a hello message is sent to RSU with nonnull TTL value, and the TTL information is updated into the queuing table of the control center. If the control center does not receive any update from the taxi after a certain period of time or the difference between the new timestamp and the old timestamp is greater than the TTL value, the control center deletes the taxi from the queuing table. Figure 2 illustrates the boundary area diagram.

3.3. Path Planning Protocol. The final part of the zone queuing scheme is path planning. To achieve path planning, the information of the taxi carrying history must be recorded and the recorded information must be classified regularly. The path can then be planned according to the records at any time and any place. In this section, the data structure of the carrying records is presented first. Next, the methods involved in collecting and integrating the carrying records are introduced. Finally, the path planning mechanism is discussed.

3.3.1. Collection and Management of Carrying Records. For collecting carrying records, each RSU maintains a carrying record table as shown in Table 1. When a taxi obtains a customer, it transmits a carrying message to the RSU, which is in its grid to notify the information of the location and time about the passenger. Because the number of the carrying records is extremely large, the records must be managed effectively to derive useful information. Thus, the carrying records were divided into seven sets according to the days of the week, and each set was divided into 24 slots.

TABLE 1: Carrying records table in RSU.

Date	Week	Time interval	Vehicular ID	Location
2014/1/12	Sat.	(17-18)	AX-15	(24.186861, 120.646382)

FIGURE 2: Queuing and boundary area diagram.

When the carrying records are classified, the hunting rate of passengers in each time slot of each day of the week must be calculated. Each RSU then transmits the set of the calculated values to the control center periodically, and the control center plans the paths for taxi drivers according to the transmitted set. The following assumptions were defined to explain the calculation efficiently:

- (1) $W = \{\text{Mon, Tue, Wed, Thu, Fri, Sat, Sun}\}$ indicates the carrying day of the week;
- (2) $T = \{(0-1), (1-2), \dots, (23-24)\}$ indicates the time interval for the carrying record;
- (3) $R_j^i = \{r_1, r_2, \dots, r_m\}$ indicates a set of carrying records in day of week $i \in W$ and time slot $j \in T$ received from t axis S_j^i indicates the number of hirings in day of week $i \in W$ and time slot $j \in T$;
- (4) H_j^i indicates the hunting rate for day of week $i \in W$ and time slot $j \in T$;
- (5) N_j^i indicates historical hunting rate in day of week $i \in W$ and time slot $j \in T$.

The following algorithm will be performed at the end of day of week i and time slot j for calculating the hunting rate, which is used in the path planning phase and operates as follows.

Hunting Rate Calculation Algorithm

- (1) $S_j^i = 0$
- (2) Read (H_j^i)
- (3) Read (N_j^i)
- (4) **Do** $S_j^i = \text{Count}(R_j^i)$
- (5) $H_j^i = (H_j^i * N_j^i + S_j^i) / (N_j^i + 1)$
- (6) $N_j^i = N_j^i + 1$
- (7) Write (N_j^i)
- (8) Write (H_j^i)
- (9) return H_j^i

(1) (Line (1)) First, variable S_j^i is initialized. It is used to store the number of carrying records of each time slot i and

FIGURE 4: System overview.

FIGURE 5: Dispatching message from queuing zone on the user interface.

to be more user-friendly. To design and implement the proposed system, this study used Xcode Version 4.4 as the integrated development environment (IDE) and Objective-C as the programming language. Notably, Xcode and Objective-C are the original IDE and programming language in iOS platform.

To handle communications between taxis and passengers, this study used IVCU, a WAVE/DSRC communication device developed by ITRI of Taiwan to provide end-to-end communications. This device currently can transmit only UDP packet to a UDP port number that is in the same subnetwork. Because it is a prototype, it is like a forwarder, not a router; therefore, two ports must be implemented in different devices as a pair before initiating the packet transmission. The prototype provides a Web interface for users to set the relation of each device.

4.3. System Prototype. The monitoring system can show all on-queue taxis on a digital map, and it can also show the queuing sequence on the left-hand side. Although the control center is operated automatically, an administrator can manage and dispatch taxis using this monitoring system if required.

Figure 5 illustrates the user interface designed for taxi drivers. When a taxi goes into a queuing zone, it can join the

TABLE 2: Statistical data table for the fixed taxi queuing point at Wuri station.

Item	Value
Average waiting time for taxi	60 (minutes)
Average number of customers per day	8 (sets)
Average working time for a driver	10 (hours)
Average number of taxis in a queuing point	76 (units)
Average income for a driver	1520 (NT dollars)

queue if it is in service. Drivers can use navigation buttons to request the control center to plan paths. Moreover, drivers can see the boundary line on the digital map, and a warning message is provided if they are too close to the switching domain. This can prevent taxis from being deleted because they cross over the domain. Drivers can also adjust some personal settings such as the priority of modes or start the navigation in the setting window. When the taxi receives dispatching messages from the control center (Figure 5), the location of the passenger, distance between the taxi and the passenger, and estimated arrival time are provided in a conforming window. The driver can decide whether to pick up the passenger. The information about the destination is not provided to prevent drivers from refusing to accept short-distance rides. If the driver accepts the mission, a navigation service is also provided to the drivers if it is required.

5. Experimental Results

This section presents the experimental results. The results include field investigation about real-taxi behavior and simulation results with investigation statistics. On this part, we use PRISM to simulate and prove that the efficiency of this model is better than waiting model in which taxi drivers wait in a fixed taxi queuing point and hunting model in which taxi drivers obtain passengers by hunting on the road.

5.1. Field Investigation. To obtain rational experimental data, this study designed a questionnaire for taxi drivers and investigated the Wuri station of the Taiwan High-Speed Rail, Taichung. Approximately 30 effective questionnaires were retrieved. As shown in Table 2, the average waiting time for a driver at the Wuri station is approximately 60 minutes, average number of customers is eight sets, and average working time is nearly 10 hours. The average income can be calculated by multiplying the average number of customers

FIGURE 6: Results of number of customers and vacancy rate.

by the income generated in the average distance. It was established according to charging standard in Taichung, and the statistical data of the average distance were provided by the Ministry of Transportation Communications.

5.2. Simulation Results of Average Number of Customers, Waiting Time, and Vacancy Rate. In this section, we simulate waiting, hunting, and our proposed zone queuing models via PRISM to evaluate models' performance. To make the simulation environment more practical, we set the experimental environment and parameter values of simulation equal to the values obtained by field investigation in Table 2.

Figure 6(a) depicts the result of the average number of customers. The zone queuing model can obtain more customers than the waiting and hunting models do because this model combines the waiting and hunting model. When drivers drive in a queuing zone, they can obtain customers from the street and can also obtain customers from the dispatching center if they wait in a queue. The hunting model is more favorable than the waiting model because taxis consume very little gas when they wait at a fixed queuing point. Therefore, most taxi drivers go to fixed queuing points, even if they must spend a longer time waiting for customers; this is because drivers perceive this model as a stable business approach.

Figure 6(b) shows the results of average waiting time for a taxi driver. Drivers obtained more customers in zone queuing mode than they did in waiting and hunting modes. The average waiting time for each taxi driver in the zone queuing mode is lower than that in the hunting and waiting

modes. They may obtain a customer in approximately 20 to 25 minutes. However, the hunting mode demonstrates more favorable performance than the waiting mode because drivers can roam in the streets according to their experience and they may have support by their companies to help them obtain customers more easily. By contrast, in the waiting mode, taxi drivers hold on positions and wait for customers; their only option is choosing a fixed queuing point that has more service requests and fewer taxis waiting. Drivers obtain customers in the hunting mode on average every 30 minutes, whereas they obtain customers in the waiting mode on average every 1 hour.

The vacancy rate is a vital indicator of the performance of each mode. Figure 6(c) shows the average vacancy rate for taxis. The vacancy rate of the zone queuing model is approximately 53%. Although it is greater than half, according to the investigation, the average vacancy rate of taxis in Taiwan is nearly 65%; therefore, the zone queuing model can reduce more than 10% vacancy for taxi drivers. Regarding the hunting mode, the vacancy rate is approximately 60%, and this rate is more consistent with the investigation; however, the waiting model has a vacancy rate of approximately 80%. This thus indicates that a driver wasted considerable time in a fixed queuing point to wait for customers again.

5.3. Simulation Results of Average Profit. The average profit was derived by analyzing the average waiting time and carrying time. The popular brand, Toyota, a taxi business brand, was used as the simulation vehicle. The average income and average cost values for taxi drivers were calculated

FIGURE 7: Results of average profit.

according to the average distance for each customer and average fuel consumption. The average profit by income and cost can thus be derived.

Figure 7(a) shows the average income for taxi drivers. As expected, the waiting model has lower income than the hunting and zone queuing modes because income is directly proportional to the number of customers. Figure 7(a) illustrates the relationship between time and the average income.

Figure 7(b) shows the average cost for taxi drivers. The values of the hunting model and zone queuing model are similar because this study assumed in the simulation that drivers are always roaming for customers either in the hunting model or the zone queuing mode. However, the greatest advantage of the waiting model is that drivers are not required to roam for customers because they are already waiting at a hot spot. Thus, they can reduce gas consumption.

Figure 7(c) shows the evaluated average profit for drivers. The average profit represents the amount of money a driver earns in a day. The zone queuing model demonstrates the highest profit among all modes because drivers in this model can hunt on the road or wait in a queue.

6. Conclusions

This study designed a geo-aware taxi carrying management system by using LBS and zone queuing techniques on Internet of vehicles, including zone queuing establishment, zone queuing management, and path planning protocol. The system can provide taxi drivers a new style to hail customers. In

this system, taxi drivers can either hunt or wait for passengers in a queuing zone, which provides an LBS in a specific place that has high demand for taxi service. This study also presents a path planning service for taxi drivers according to the history carrying records, which enhance hunting efficiency and improve service quality. PRISM was used to perform our simulation, and the results show that the proposed system has more favorable performance than existing systems. In the future, zone queuing and cruising mechanisms will be developed according to hot spot information, grid weight values, and crowdsourcing information and a prototype of the zone queuing and cruising system will be implemented to provide advice and guidance to taxi drivers. Also, we will use Big Data resources provided by Taiwan Taxi Company, the biggest taxi company in Taiwan, to analyze LBS data and calculate average waiting time of each queuing zone and then adaptively guide taxis to different hot spots.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

The authors would like to thank the Ministry of Science and Technology of the Republic of China for financially supporting this research under Contract nos. MOST103-2221-E-035-057 and MOST104-2221-E-035-021.

References

- [1] R. He, A. F. Molisch, F. Tufvesson, Z. Zhong, B. Ai, and T. Zhang, "Cooperative positioning for vehicular networks: facts and future," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2237–2248, 2014.
- [2] P. Papadimitratos, A. La Fortelle, K. Evenssen, R. Brignolo, and S. Cosenza, "Vehicular communication systems: enabling technologies, applications, and future outlook on intelligent transportation," *IEEE Communications Magazine*, vol. 47, no. 11, pp. 84–95, 2009.
- [3] J. Li, T. Luo, and L. Ding, "A location aware based handoff algorithm in V2I system of railway environment," in *Proceedings of the Global Information Infrastructure and Networking Symposium (GIIS '14)*, pp. 1–6, IEEE, Québec, Canada, September 2014.
- [4] A. Singhrova and N. Prakash, "Vertical handoff decision algorithm for improved quality of service in heterogeneous wireless networks," *IET Communications*, vol. 6, no. 2, pp. 211–223, 2012.
- [5] K.-H. Chen, C.-R. Dow, S.-C. Chen, Y.-S. Lee, and S.-F. Hwang, "HarpiaGrid: a geography-aware grid-based routing protocol for vehicular Ad Hoc networks," *Journal of Information Science and Engineering*, vol. 26, no. 3, pp. 817–832, 2010.
- [6] PRISM, <http://www.prismmodelchecker.org>.
- [7] Y. J. Lee, J. Trevathan, I. Atkinson, W. Read, T. Myers, and R. Johnstone, "The evolution of the SEMAT sensor network management system," in *Proceedings of the 7th International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pp. 229–234, IEEE, Adelaide, Australia, December 2011.
- [8] N. Alam and A. G. Dempster, "Cooperative positioning for vehicular networks: facts and future," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1708–1717, 2013.
- [9] P. J. Tseng, C. C. Hung, T. H. Chang, and Y. H. Chung, "Real-time urban traffic sensing with GPS equipped probe vehicles," in *Proceedings of the 12th International Conference on ITS Telecommunications (ITST '12)*, pp. 306–310, Taipei, Taiwan, November 2012.
- [10] C. T. Barba, M. Á. Mateos, P. R. Soto, A. M. Mezher, and M. A. Igartua, "Smart city for VANETs using warning messages, traffic statistics and intelligent traffic lights," in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV '12)*, pp. 902–907, IEEE, Alcalá de Henares, Spain, June 2012.
- [11] A. Dua, N. Kumar, and S. Bawa, "A systematic review on routing protocols for vehicular Ad-Hoc networks," *Vehicular Communications*, vol. 1, no. 1, pp. 33–52, 2014.
- [12] D. Lee, Y. Kim, and H. Lee, "Route prediction based vehicular mobility management scheme for VANET," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 679780, 9 pages, 2014.
- [13] X.-Y. Bai, X.-M. Ye, H. Jiang, and J. Li, "A novel traffic information system for VANET based on location service," in *Proceedings of the 16th International Conference on Networks (ICON '08)*, pp. 1–6, New Delhi, India, December 2008.
- [14] S. M. Tornell, C. T. Calafate, J. C. Cano, and P. Manzoni, "Implementing and testing a driving safety application for smartphones based on the eMDR protocol," in *Proceedings of the IFIP Wireless Days (WD '12)*, pp. 1–3, Dublin, Ireland, November 2012.
- [15] R. Tiwari and N. Kumar, "Minimizing query delay using co-operation in IVANET," *Procedia Computer Science*, vol. 57, pp. 84–90, 2015.
- [16] C. Chen, D. Zhang, N. Li, L. Sun, S. Li, and Z. Wang, "iBOAT: isolation-based online anomalous trajectory detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 806–818, 2013.
- [17] S.-F. Cheng and T. D. Nguyen, "TaxiSim: a multiagent simulation platform for evaluating taxi fleet operations," in *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT '11)*, pp. 14–21, IEEE, Lyon, France, August 2011.
- [18] P.-Y. Chen, J.-W. Liu, and W.-T. Chen, "A fuel-saving and pollution-reducing dynamic taxi-sharing protocol in VANETs," in *Proceedings of the 72nd Vehicular Technology Conference Fall (VTC '10)*, pp. 1–5, IEEE, Ottawa, Canada, September 2010.
- [19] J.-P. Sheu, G.-Y. Chang, and C.-H. Chen, "A distributed taxi hailing protocol in vehicular ad-hoc networks," in *Proceedings of the IEEE 71st Vehicular Technology Conference (VTC '10)*, pp. 1–5, IEEE, Taipei, Taiwan, May 2010.
- [20] H. E. Hosni, N. Farhat, R. Nimer et al., "An optimization-based approach for passenger to shared taxi allocation," in *Proceedings of the 20th International Conference on Software, Telecommunications and Computer Networks (SoftCOM '12)*, pp. 1–7, IEEE, Split, Croatia, September 2012.
- [21] N. Liu, M. Liu, J. Cao, G. Chen, and W. Lou, "When transportation meets communication: V2P over VANETs," in *Proceedings of the 30th IEEE International Conference on Distributed Computing Systems (ICDCS '10)*, pp. 567–576, Genova, Italy, June 2010.
- [22] Y. Yuan, Y. Wu, W. Zheng, and K. Li, "Guarantee strict fairness and utilize prediction better in parallel job scheduling," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 4, pp. 971–981, 2014.
- [23] N. McKeown, "The iSLIP scheduling algorithm for input-queued switches," *IEEE/ACM Transactions on Networking*, vol. 7, no. 2, pp. 188–201, 1999.
- [24] V. Gabale, B. Raman, P. Dutta, and S. Kalyanraman, "A classification framework for scheduling algorithms in wireless mesh networks," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 1, pp. 199–222, 2013.
- [25] M. Fan and G. Quan, "Harmonic-aware multi-core scheduling for fixed-priority real-time systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 6, pp. 1476–1488, 2014.
- [26] S. Ren and M. van der Schaar, "Dynamic scheduling and pricing in wireless cloud computing," *IEEE Transactions on Mobile Computing*, vol. 13, no. 10, pp. 2283–2292, 2014.

Hindawi

Submit your manuscripts at
<http://www.hindawi.com>