GeoMob: A Mobility-aware Geocast Scheme in Metropolitans via Taxicabs and Buses

Lei Zhang, Boyang Yu, Jianping Pan
University of Victoria, Victoria, BC, Canada

***Abstract*—Geocast, delivering messages to a specific location, has become an important issue with the accelerated development of the location-based services in mobile networks. Geocast in the automotive domain is of particular interest, enabling many promising applications, such as geographic advertising, location-based traffic alerts, etc. Different from the conventional geocast algorithms focusing on the distance-based approaches, in this paper, we propose a mobility-aware geocast algorithm (GeoMob) for urban VANETs from the Delay-Tolerant Network (DTN) perspective to better deal with the high mobility and transient connectivity issues. Different levels and aspects of vehicle mobility information are employed, making GeoMob very simple, scalable and communication and compunction-effective. Practical issues are well considered by introducing real-world trace analysis, trace-driven simulation and efficient buffer management. Extensive performance comparisons with other protocols have been conducted to show the advantages of GeoMob.**

***Index Terms*—Geocast, VANETs, mobility, DTNs**

I. INTRODUCTION

In mobile networks, geocast forwards messages to destinations with specific geographic locations. The existing work of geocast mainly focuses on geometric distance-based approaches [1]–[4], where the distance to the destination is taken as the most important criterion for routing. However, they are not suitable for the large-scale urban Vehicular Ad-hoc NETwork (VANET) for the following reasons: first, because of the high node mobility and complex road structure, the distance relations of nodes change frequently and quickly, causing the reduction in the performance of the distance-based schemes; second, the schemes which require network topology information such as GeoTORA [1] and GeoGrid [2], need to frequently update their knowledge of the network, which can cause tremendous overhead in a large-scale network, such as the urban VANET with thousands of nodes; third, none of the existing work takes into account the node mobility, which has a strong correlation with the geographic locations and geocast.

On the other hand, routing in Delay-Tolerant Networks (DTNs), which specializes on intermittent connectivity, is extensively studied [5]–[7]. We find that, even not designed for geocast, many existing DTN routing schemes can adapt to geocast with stationary destinations. However, these DTN routing schemes are originally designed for relatively small-scale networks, e.g., with up to hundreds of nodes. Concerning about the scale, existing schemes either fail to achieve an acceptable performance due to the flooding-like mechanisms [5], or introduce enormous communication and computation overhead, such as pair-wise node contact history information maintenance [6], [7].

Instead of the traditional geometry-based approaches, we propose a mobility-aware geocast scheme (GeoMob) for large-scale urban VANETs from the DTN perspective, through Vehicle-to-Vehicle (V2V) communication. Different from some of the most efficient DTN routing schemes [6], [7], which are based on the expensive pair-wise contact probability calculation and sharing, our scheme employs the node mobility information at different levels, i.e., macroscopic and microscopic mobility. Macroscopic mobility describes the traffic trend of all vehicles in a city, while microscopic mobility captures the mobility patterns of individuals. Because the macroscopic mobility for a city is quite stable and the microscopic mobility is completely self-maintained by each vehicle. This mobility hierarchy makes our scheme extremely simple, scalable and communication and computation-efficient when compared with existing solutions.

The two levels of mobility are extracted from the real-world GPS traces of taxicabs and buses in Shanghai, China. To facilitate the mobility modeling, we divide the city into regions, each of which contains considerable traffic volumes. Traffic flows among regions are extracted and utilized as the macroscopic mobility pattern. The volume of the traffic flows can indicate how well the regions are connected through vehicles and how reliable the message dissemination between regions can be via vehicular communication. The massive data trace also allows us to investigate each individual's mobility pattern, which serves as the routing criterion. The proposed scheme also employs the different mobility features of different vehicle types, i.e., taxicabs and buses. Considering practical restrictions, i.e., the limited buffer size and transmission bandwidth, an efficient buffer management is introduced which further improves the performance of our scheme.

The rest of the paper is organized as follows. In Section II, conventional geocast and related DTN routing schemes are discussed as related work. The trace analysis is presented in Section III. Section IV reveals the details about the proposed geocast scheme. With a trace-driven simulation, the performance is evaluated in Section V. Finally, we conclude our paper in Section VI.

II. RELATED WORK

A. Traditional Geocast Schemes

The study of geocast has a relatively long history since 1987 [8]. In [1], [8], only unicast is considered. To further improve the delivery ratio, multicast, e.g., directed flooding, is widely adopted by many schemes. [3] proposed two schemes

978-1-4799-3360-0/14/$31.00 ©2014 IEEE

978-14799-3360-0/14/$31.00 ©2014 IEEE

based on directed flooding, where one defines a rectangular forwarding zone between the source and destination and the other forwards messages to all the neighbors who have shorter distances to the destination. [9] introduced Voronoi diagram for the forwarding zone selection. Another category of geocast schemes uses group-based approaches. In GeoGrid [2], the network is partitioned into logic grids and each partition selects one single gateway node as a group representative to forward messages. [4] introduced additional infrastructures as the gateways, collecting data from mobile nodes and forwarding them to the destination. The maintenance of gateways introduces extra overhead. [10], [11] deploy navigation system for geocast purposes, causing additional requirement on software.

B. DTN Routing Schemes

Delay-Tolerant Networks (DTNs) enable communications where the source to destination connectivity cannot be always sustained. VANET is a typical delay-tolerant network. Compared with the conventional geocast algorithms, DTN routing is more capable of dealing with high node mobility and transient node connectivity. For such reasons, we propose a geocast solution from the DTN's point of view in this paper.

Besides flooding-based schemes [5], [12], another very important DTN routing category is the contact information-based routing, where a smarter relay node selection is made. In Prophet [6], each node maintains the encounter history with other nodes, and the routing decision is made based on the encounter probability. MaxProp [7] also utilizes the encounter information to estimate the cost of a virtual end-to-end path to the destination and uses it as the metric for routing decisions. MaxProp also takes into account realistic issues such as buffer size and bandwidth limitation, which is similar to our work. However, in [7], MaxProp is only tested on bus traces. [13], [14] further take into account the inter-contact time, and thus are much more complicated.

III. Trace Analysis

The proposed work is based on real vehicle GPS traces collected in a modern city, Shanghai, China. There are two main benefits of introducing the real trace. First, analyzing real-world traces enables us to better understand the overall geographic distribution of the city traffic (macroscopic mobility) and individual vehicle mobility patterns (microscopic mobility). Introducing both levels of mobility into the geocast scheme is the most significant novelty and contribution of our work. Second, the real-world traces enhance the reliability of our scheme by providing the realistic user mobility.

A. About the Traces

The traces we used (partially available at http://www.cse.ust.hk/scrg) were collected from vehicles in Shanghai, i.e., $2,299$ taxicabs from Jan. 31, 2007 to Feb. 27, 2007 and $2,500$ buses of 103 routes from Feb. 24, 2007 to Mar. 27, 2007. Each bus reported a GPS report every one minute while taxicabs reported every 15 seconds if there was no customer on board and every one minute when with customers. The information contained in the trace includes the vehicle ID, the latitude and longitude location, timestamp, vehicle moving speed and heading direction. In addition, taxicabs also reported whether they are hired by customers.

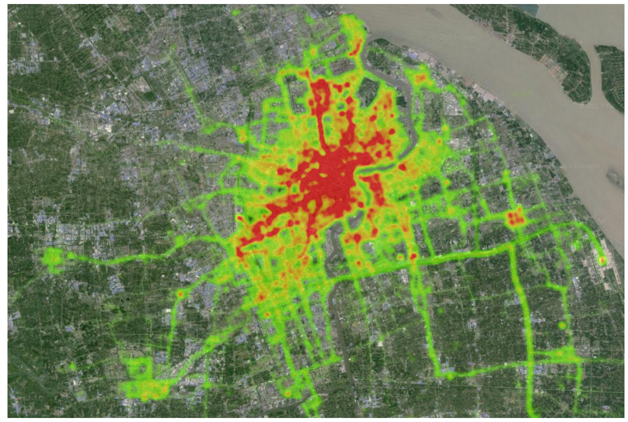


Fig. 1: Traffic heat.

Different from other work taking account of only one type of vehicles in the network, in this paper, we adopt the collaboration of the different types of vehicles, i.e., taxicabs and buses. This increases the diversity of node mobility patterns, which can benefit the message dissemination for geocast. The mobility patterns considered are summarized as follows:

a) Buses: Each bus has a limited spatial and temporal coverage, i.e., moving along fixed routes during a certain period of time, which implies a very strong mobility pattern. And this can be very helpful for geocast since the node mobility is somehow predictable. Given a message destination area, the buses whose routes cross that area should be preferred as message forwarders.

b) Taxicabs: Compared with buses, the pattern of taxicab mobility is more diverse with a much larger spatial and temporal coverage. By intuition, the taxicab mobility is impacted by two factors: 1) customer demands and 2) taxicab drivers' driving habits. If one taxicab is occupied by customers, the mobility is mainly determined by the customer destination. The driver may pick the shortest path or a path with least congestion. If the taxicab is not occupied, the mobility depends on the taxi driver's driving habits and "hunting" preference.

The traces cover a large portion of Shanghai city, with the horizontal span of 70 km from the west-most record point (Hongqiao Airport) to the east-most one (Pudong Airport), and 45 km vertical span from the north-most point (Baoshan District) to the south-most one (South of Minhang District). Such a square area has a size of around $3,150$ km^2. However, it is not all spread with vehicles because of the city and road structures. Public vehicles appear more often in hot social spots, such as transportation hubs, commercial areas, and those regions connecting the hot spots. By counting the number of the GPS records of all taxis, we plot Fig. 1 which shows the traffic condition of the city. Red color indicates a high traffic volume while green color indicates a lower volume.

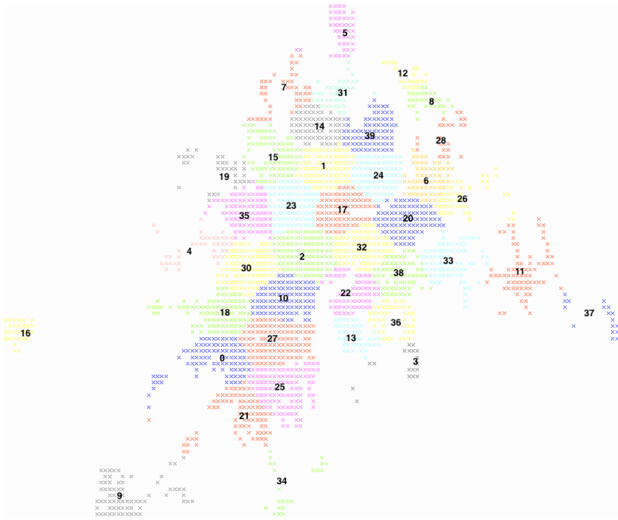


Fig. 2: Clustered regions.

B. Trace Preprocessing and Region Clustering

Due to the large scale of the map, it is hard to outline all details of the regions which have distinct traffic volumes. Thus, we first discretize the map as a tiling of square cells with a size of 1 km $*$ 1 km. We let each GPS location of a report be represented by the cell where it belongs to. Then each vehicle trajectory trace is converted to a sequence of cells. We focus on the cells with a considerable amount of traffic by counting its GPS report frequency. The cells, where taxi drivers pick up or drop off customers, are also what we are interested in.

With the focused cells identified, we cluster them into regions. And it helps to describe vehicle mobility in a proper granularity concisely. We apply the k-means clustering algorithm to these cells. The value of k determines how many regions can be formed. Since the total number of cells is fixed, with a larger value of k, more regions will be formed, but with a smaller size each. Notice that clustering is only one method to define the geographic regions, i.e., the size, location and shape. For generality, we adopt the clustering algorithm to generate regions with similar sizes and set 40 as our default number of regions.

Traditional clustering algorithm is based on the Euclidean point distance but we use the travel distance of two locations instead. This is because we take the real-world road structure into consideration to reflect the actual reachability between any two locations. Travel distances can be obtained through online map services, e.g., Google maps. The clustering result is shown in Fig. 2, where all regions are numbered and colored.

C. Two-level Mobility Mining

For mobile ad-hoc networks, node mobility plays a significant role in opportunistic forwarding-based routing protocols. Especially for geographic routings such as geocast, a proper understanding and utilization of the node mobility can help greatly improve the performance. Two levels of mobility patterns, macroscopic and microscopic, are extracted from the real-world vehicle traces.

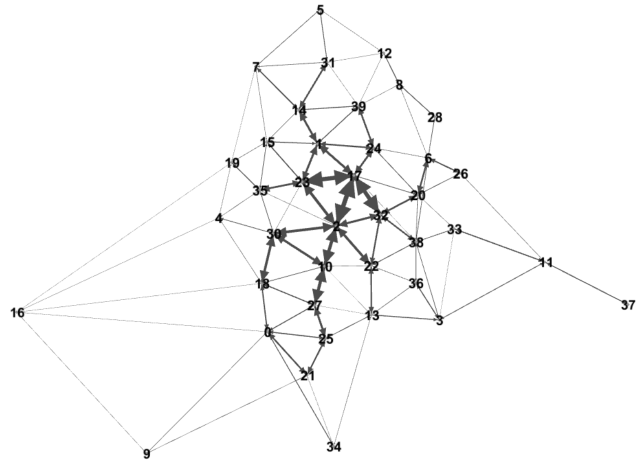
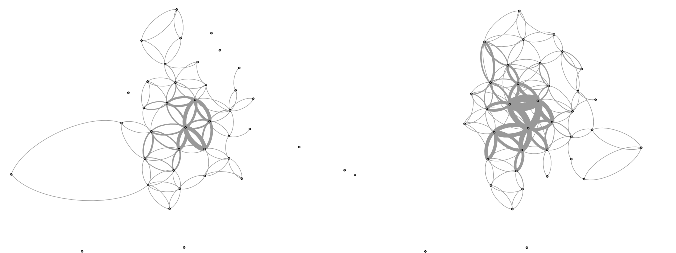


Fig. 3: Macroscopic mobility patterns.



(a) Taxi 0094

(b) Taxi 01292

Fig. 4: Microscopic patterns for individual taxis.

1) Macroscopic mobility pattern: Macroscopic mobility pattern reflects the overall traffic condition. It can be either provided by the urban planning or transportation department or obtained by counting the number of vehicle commutes between regions. More of such vehicle commutes imply the stronger traffic flow between the regions, which further implies that higher reliability to transfer messages from one region to the other via vehicles. The macroscopic mobility characterizes the traffic flows between any pair of neighbor regions in the city. It reflects how regions are connected by vehicle traffic and how strong each connection is. We express the macroscopic mobility pattern by a weighted directed graph, $MacMP(\mathbf{V}, \mathbf{E}, \mathbf{w})$, as shown in Fig. 3, where vertices, $\mathbf{V}$, represent the regions shown in Fig. 2, the directed edges, $\mathbf{E}$, represent the traffic flows and the thickness, $\mathbf{w}$, of edges represents the amount of the vehicle traffic, i.e., the strength of the connection. We can observe the strong region connections in downtown areas, i.e., region 2, 17, 23 and 32, and weaker connections on the periphery of the city, i.e., airports (region 16 and 37). Another property of the macroscopic mobility is that such patterns are relatively stable for the whole city during different time periods of a day and different days of a month according to our observation. This is an attractive feature as such information does not need a frequent update, which greatly reduces the scheme complexity.

2) Microscopic mobility pattern: Microscopic mobility pattern, on the other hand, captures the motion patterns of individual vehicles. For buses, the ones which belong to different routes, have different mobility coverage. For taxis, individuals have different mobility pattens caused by different driving behaviors of the drivers, e.g., some drivers prefer to work in downtown areas while others prefer to take longer-distance businesses, such as to airports. The mobility pattern can also be shown using weighted graph similar to Fig. 3. Figure 4(a) and Fig. 4(b) depict the patterns of two taxis over the data collection period. An obvious difference could be observed, i.e., Taxi 0094 was active in only downtown areas while Taxi 01292 showed more activities in more regions. For a specific vehicle v, the microscopic mobility pattern can be presented as a set of conditional probabilities,

$$MicMP_v = \bigcup P(f_i|h_n h_{n-1} \cdots h_1 h_0), f_i, h_j \in \mathbf{V}, n \leq N,$$

which records all the transfer probabilities to certain regions given the history information as a sequence of regions. Here f_i indicates the region which is possible for v to go to, given that it has come from regions $h_n, h_{n-1}, \cdots, h_1, h_0$ and h_0 indicates the current region of the vehicle. If the history contains the n previous regions, we call it nth-order conditional probability. Given a threshold N, we represent each vehicle's microscopic mobility pattern with all its possible nth-order conditional probabilities where $n \leq N$.

A very desirable feature is that the microscopic mobility pattern is totally self-maintained by a vehicle itself because it only depends on its movement. No information sharing between vehicles is needed. The process of the conditional probabilities updating is described as follows. Suppose vehicle v has come from region sequence $H_n = h_n, h_{n-1} \cdots h_1$ and is currently in region h_0, it is possible for it to go to one of the neighbor regions, i.e., future region set $\mathbf{FR} = \{f_1 \cdots f_m\}$. And finally it goes to f_1. Then it updates its $MicMP$ as follows:

$$P(f_1|H_n)_{new} = P(f_1|H_n)_{old} + (1 - P(f_1|H_n)_{old}) * P_{init},$$

and for the other regions, e.g., region f_x in $\mathbf{FR}$ except f_1, which are not chosen,

$$P(f_x|H_n)_{new} = P(f_x|H_n)_{old} - \frac{(1 - P(f_1|H_n)_{old}) * P_{init}}{m-1}.$$

P_{init} is a scale factor between 0 to 1.

D. Mobility-Entropy

To a large extent, our proposed routing scheme depends on the microscopic mobility patterns of individual taxis. It is important to understand how strong the microscopic mobility patterns are for different vehicles and how they change over time during a normal work day. To quantitively show the activeness of individual mobility, we introduce the mobility-entropy. An example is given as follows.

With clustered regions, the trip of a taxi during a certain time period can be represented as a sequence of geographic regions, e.g., for taxi A and B,

$$T_A = r_3, r_2, r_2, r_3, r_5, r_2,$$
$$T_B = r_1, r_2, r_3, r_5, r_4, r_1.$$

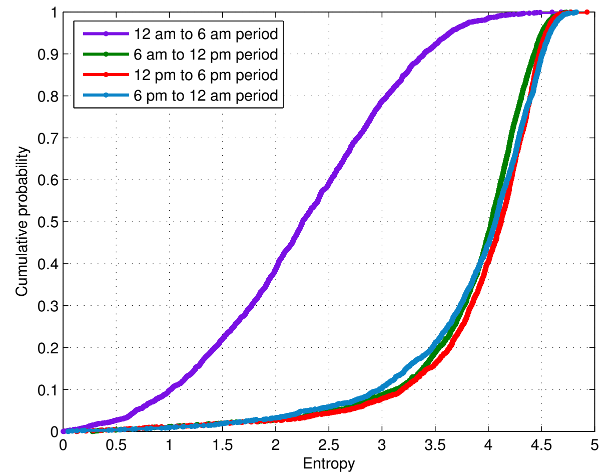


Fig. 5: Entropy distributions.

During this short period, for taxi A, the visiting frequencies of regions r_3, r_2, r_5 are $\frac{2}{6}, \frac{3}{6}, \frac{1}{6}$, respectively. And for taxi B, the visiting frequencies of regions r_1, r_2, r_3, r_4, r_5 are $\frac{2}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}$, respectively. We can observe that taxi A has a narrower mobility pattern as it moves in a limited number of regions, i.e., only three regions. On the other hand, the trace of B has higher randomness with 5 regions. Thus, introducing the similar concept of entropy in communication theory, the mobility-entropy can be calculated:

$$E_A = -\frac{2}{6}\log\frac{2}{6} - \frac{3}{6}\log\frac{3}{6} - \frac{1}{6}\log\frac{1}{6} = 0.439,$$
$$E_B = -\frac{2}{6}\log\frac{2}{6} - \frac{1}{6}\log\frac{1}{6} * 4 = 0.678.$$

Taxi A has a more predictable pattern than B, which can be shown as $E_A < E_B$. We statistically studied the trace of all taxis over the whole data collection period. With different time granularity, we are able to obtain the activeness of a taxi at time periods with different lengths. For simplicity, we divide a day, i.e., 24 hours, into equal-length periods, saying two 12-hour periods scenario, three 8-hour periods scenario and four 6-hour periods scenario. We compare the mobility-entropy distributions of different periods in each scenario. We have found that the most distinct difference of different periods appears in the 6-hour period scenario, as shown in Fig. 5. The following conclusions can be obtained: 1) For the first 6-hour period, from 0 to 6 am, the entropies are much smaller than those of other periods, implying taxis are a lot less active during this period and the mobility pattern is very deterministic. 2) The entropy distributions of the other three periods are extremely similar, meaning the activeness of the taxis during their usual work hours, i.e., 6 am to 12 pm, is very similar. 3) From 6 am to 12 pm, the majority (70%) of taxis has an entropy falling between 3.5 to 4.5, showing that most taxis have a similar activeness during that period of time, although the individual difference exists as shown in Fig. 4. The mobility-entropy study provides us more insights of the taxicab mobility pattern, i.e., how strong the patterns

TABLE I: Notations

Notation	Explanation
$\mathbf{FR}$	A list of a vehicle's future regions
$\mathbf{TG}$	A list of a message's ToGo regions
M_i	Message with ID i
L_i^j	Vehicle j's forwarding likelihood of message i
N	Threshold for the number of history regions
$h_n h_{n-1} \cdots h_1 h_0$	The nth order history regions of a vehicle

are, how they change over time and how the pattern strength is distributed over all taxis. This is an important supplementary study to enhance our understanding of the vehicle mobility, which is the key of the proposed scheme.

IV. Geocast Routing Scheme

In this section, we propose a geocast scheme in a city scenario via taxicabs and buses. As a basic assumption in VANETs, all vehicles are cooperative for message transmission. Generated from the source node, a message is first forwarded towards the destination area through multiple hops. Once it reaches the target area, it is simply flooded within that area. Because the message flooding within the target area is relatively simple, in this paper, we mainly focus on the geographic routing from the source to the target area, which is in a unicast fashion. This is different from the conventional unicast where the destination is a specific node. The message forwarding scheme consists of two parts: message forwarding strategy and buffer management. The two levels of mobility pattens extracted are utilized in message forwarding strategy where macroscopic mobility pattern is used in the forwarding path selection and microscopic mobility patterns are used in routing decision making.

A. Message Routing Strategy

The key idea of GeoMob message routing is to utilize the two levels of vehicle mobility. An optimal routing path, which is a sequence of regions towards the destination, is selected based on the macroscopic mobility when a message is generated. Then vehicles try to forward this message right along such path towards the destination. A vehicle's microscopic mobility determines whether it can help to forward the message to its next hop on the optimal path.

1) Macroscopic mobility-based forwarding path selection: It is believed that the larger traffic volume towards a region implies a better connectivity to that region. The chance of successfully delivering a message to that region is also higher. Thus we start our geocast routing by first selecting an optimal forwarding path leading to the destination with the best connectivity based on the graph shown in Fig. 3. Once a message is generated, the source calculates the optimal path leading to the destination. Such an optimal path is a region sequence which has the best connectivity from the origin to the target destination and the path information is embedded in the message header once the path is calculated. To find such a path, Dijkstra algorithm is applied to the weighted graph in Fig. 2, where the weight on the edge is determined by the total traffic volume between regions. All vehicles should pre-load the overall traffic weight graph to be capable of calculating the optimal path for each message. Since the city traffic pattern does not change dramatically over time, there is no need to frequently update this graph, which greatly reduces the overhead for vehicles. Vehicles try their best to forward the messages along their optimal paths, to increase the delivery ratio. The message is transferred in a region-by-region manner, i.e., once the message enters one region, suitable vehicles are selected to forward it to the next region along the optimal path.

2) Microscopic mobility-based routing decision: Each vehicle also maintains its own mobility pattern, i.e., the microscopic mobility pattern. Individual mobility pattern is used for making explicit routing decision when two vehicles encounter with each other. As mentioned in Section III, such a pattern is represented as a collection of conditional probabilities. These conditional probabilities can be obtained from the movement history and further updated as vehicles move among the regions. The updating of microscopic mobility patterns is presented in Section III.

According to the current vehicle location, relay vehicles will try to forward a message to the next optimal region according to the optimal path stored in the message. We use the term *likelihood* to determine how capable a vehicle is to forward a message to the next optimal region. Assume the optimal path for message m is $r_2 \rightarrow r_3 \rightarrow r_7 \rightarrow r_5 \rightarrow r_6$, where r_6 indicates the destination region. When two vehicles encounter with each other in region r_5, one of them, say v_i is carrying m and the other, v_j, is not. Then v_i first requests v_j's message delivery likelihood L_m^j for message m. If $L_m^j > L_m^i$, v_i forwards the message to v_j, otherwise, v_i does not forward. Such process is expressed in Algorithm 1.

Again, a vehicle maintains a likelihood for each message it is carrying, and these likelihoods are used to compare with other vehicles' likelihoods to make the routing decision. Algorithm 2 gives the details of how the likelihood is calculated and updated. In Algorithm 2, we utilize the near-future location information if it is available. This is because we consider the mobility features of city buses and taxis, whose near-future location may be easily accessible. We denote the near-future regions of a vehicle, if available, as "future regions" $\mathbf{FR} = \{fr_1, fr_2, \cdots, fr_n\}$ and the regions the message still has to go through according to its optimal path as "ToGo" regions $\mathbf{TG} = \{tg_1, tg_2, \cdots, tg_n\}$. Algorithm 2 can be explained as follows.

If the encountered vehicle's near-future information is known, such information is utilized. For taxis carrying customers, their destinations are determined by the customers on board. For buses, because they have pre-fixed routes, so their future regions over a short period are predictable. For such vehicles, we only need to see if their future locations have any intersections with the message's desired future optimal regions. If yes, these vehicles are definitely helpful to forward that message. Otherwise, the message is not passed to them. There are also vehicles whose destinations are unknown, such as taxis without customers. At this point, the taxi's microscopic mobility pattern becomes the source for location prediction.

According to v_j's historical mobility information, it replies to v_i with the probability of going to the optimal region given the current trajectory history, as v_j's forwarding likelihood, according to the conditional probability from its microscopic mobility pattern. If v_j has a high enough probability, i.e., higher than v_i's likelihood, to go to any of the future optimal regions of the message, v_i will forward the message to v_j. Otherwise, that specific message will not be forwarded.

Algorithm 1 Routing Strategy

procedure ROUTINGSTRATEGY(vehicle E, R and message m)
  $L_m^E = LikelihoodUpdate(E, m)$
  $L_m^R = LikelihoodUpdate(R, m)$
  if $L_m^E < L_m^R$ **then**
    if $L_m^E > 0$ **then**
      Keep m
    else
      Drop m
    end if
    if $L_m^R > 0$ **then**
      E forwards m to R
    else
      E does not forward m to R
    end if
  else
    E does not forward m to R
    if $L_m^E > 0$ **then**
      Keep m
    else
      Drop m
    end if
  end if
end procedure

Algorithm 2 LikelihoodUpdate

procedure LIKELIHOODUPDATE(vehicle V and message m)
  if V's $\mathbf{FR}$ is known **then**
    if V's $\mathbf{FR}$ intersects with m's $\mathbf{TG}$ **then**
      Find the intersection region I closest to the destination
      Record I's index in $\mathbf{TG}$ idx
      $L_m^V = 1 + \frac{idx}{TG.length}$
    else
      $L_m^V = -1$
    end if
  else
    Start nth order prediction, $n = N$
    while nth order pattern does not exist in pattern set **do**
      $n = n - 1$
      if $n < 2$ **then**
        $L_m^V = -1$
        Break
      end if
    end while
    $L_m^V = P(TG_1 | h_n h_{n-1} \cdots h_1 h_0)$
  end if
end procedure

B. Buffer Management

To be realistic, we assume each vehicle carries a limited-size buffer on board and has limited wireless transmission bandwidth. Thus when two vehicles encounter, not all the messages in the buffer can be transmitted at one contact. To effectively utilize the buffer space and each contact opportunity, the buffer management of our routing scheme involves the following mechanisms:

First, whenever a vehicle enters a new region, it updates the likelihoods for all its messages and drops those messages with low likelihoods, shown in Algorithm 1. For each message m and vehicle V pair, L_m^V changes with the location of the vehicle. When the vehicle moves towards the optimal regions of the message, L_m^V could be high; however, once the vehicle is not helpful to deliver the message, it deletes the message to reduce the extra storage and communication overhead.

Second, each vehicle prioritizes the messages in the transmission queue. When vehicle v_i encounters v_j, v_i may have more than one message suitable to be forwarded to v_j. Before the transmission starts, v_i first sorts the messages in its transmission queue according to the likelihood of L_m^j descendingly. Note that the messages are sorted according to the message delivery likelihood from v_j's point of view, so that the message which can take the most benefit from v_j gets first transmitted to v_j. This greatly increases the efficiency of each transmission.

Third, the acknowledgements of the received message are flooded in the network to remove the redundant copies of the message. We borrowed a similar idea from MaxProp [7], where each acknowledgement is a 128 bits hash of the received message ID, source and destination. The cost is little if the acknowledgment is small compared to data packets, which is 512 Kbits in our simulation setting. In addition to the small buffer size overhead, it also costs little communication overhead, which has been proved in [7] that no more than 1% of the average connection duration is spent on sending acknowledgments.

V. PERFORMANCE EVALUATION

A. Protocol Comparisons

We compare our scheme with three other DTN-based geocasting protocols and one traditional distance-based protocol. Although the DTN-based ones are not originally designed for geocast, by setting the destination with stationary mobility, we can easily convert them for geocast purpose. To distinguish from the original schemes, we add a "Geo" prefix to the original scheme names. Short descriptions about the schemes are given as follows:

- **GeoEpidemic [12].** Whenever two vehicles encounter with each other, they exchange as many messages as they have, which is a simple flooding.
- **GeoProphet [6].** Prophet makes routing decisions based on the history contacts information. Messages are forwarded to the nodes who have a higher contact frequency with the destination. A *transitive* property is considered when updating the contact frequency: if A meets B frequently and B meets C frequently, then it is believed that A and C should have a relatively high contact frequency.
- **GeoMaxProp [7].** Similar to Prophet, MaxProp also utilizes the history contact information. Each node maintains

TABLE II: Comparison among multiple schemes

Routing algorithm	Knowledge for routing decision	No. of copies	Buffer size	Bandwidth
GeoEpidemic [12]	None	Unlimited	Not mentioned	Not mentioned
GeoProphet [6]	Encounter probability	Controlled	Not mentioned	Not mentioned
GeoMaxProp [7]	Encounter probability	Controlled	Limited	Limited
GeoDist	Distance to the destination	Controlled	Not mentioned	Not mentioned
GeoMob	Mobile nodes mobility	Controlled	Limited	Limited

a contact graph based on the history contact information of both its own and its encounters. Routing decisions are made according to the cost of a delivery path going through a specific neighbor. A low-cost delivery acknowledgement is adopted and the transmission queue is prioritized considering both message hop counts and delivery probabilities.

- **GeoDist.** We have also compared our scheme with distance-based geocast scheme, which is the main algorithm for many existing geocast schemes. Messages are forwarded to the nodes with a shorter distance to the destination area. Similar to [15], the messages in the transmission queue are prioritized according to the "improvement" which can be possibly achieved by a transmission. The "improvement" here means the reduction of the distance to the destination. The greater the "improvement" the higher the priority assigned and the sooner the message can be transmitted.

A scheme comparison is shown in Table II. Flooding based schemes, i.e., Epidemic, do not make any routing decisions to select relay nodes and the amount of message copies is unlimited. The other schemes carefully select relay nodes based on different information, i.e., encounter information or distance information, leading to a controllable amount of copies for each message. The amount of the copies varies in different situations. For instance, using Prophet, if the destination is active and has rich contact history with many nodes, then a large number of copies are expected to be forwarded to those nodes. However, if the destination is not very popular, then a limited number of copies will be generated. Among all schemes, our scheme, as well as GeoMaxProp and GeoDist, has taken into account the practical restriction of buffer size and transmission bandwidth, which is reflected by introducing the buffer management mechanisms.

We provide a deeper comparison between other schemes and ours. First, the traditional distance-based geocast routing has limitation in a large-scale urban vehicular network. With the complex road structure and high vehicle mobility, the relative position of vehicles changes quickly. A vehicle, which is currently closer to the destination, may be farther in the next second, because it either takes a detour towards another area or it is moving in the opposite direction, or it just stops. Thus taking the distance as the only routing criterion is not sufficient and suitable for the city vehicle networks environment.

Second, compared with routing schemes such as Prophet and MaxProp, our scheme requires much less information from other nodes. As described, in Prophet, the routing decision for one node is based on its contact history with the destination. To make such a contact information updated, one node has to keep requesting its neighbors for their contact information with destination, whenever there is a chance. Similarly, MaxProp requires to share node's contact information among all nodes to maintain the contact graph up to date. Remember that the contact information is always changing as nodes move and meet each other, thus an enormous communication overhead will be introduced, not to mention if the network scale is large with thousands of nodes, i.e., in the city case. Managing and updating all the contact information in the local buffer, e.g., frequently searching, updating, etc., also introduces computation overhead. Most existing work did not count it as the scheme overhead since it is on the control level. However, the complexity issue can become severe in the real world.

Our scheme also requires global information, i.e., the macroscopic mobility pattern, however, the city macroscopic mobility pattern stays quite stable over time according to our observation of the trace. Unless there are some major events happening, e.g., a new highway is built which changes the city traffic, there is no need to frequently update the macroscopic mobility pattern. The other information needed in our scheme is the microscopic mobility pattern, which is totally locally-maintained by each node as described in Section III, which is a very promising feature for a large-scale network.

B. Simulation Setup

We use an open-source simulator ONE [16] for simulation. Due to the limitation of our testbed computation capability, we randomly select 200 taxis and 300 buses, importing their traces for node mobility. According to the specifications of IEEE 802.11p [17], [18], we set the vehicle transmission range to 500 m and the transmission speed to 6 Mbps. The buffer size is set to 500 MB. Considering a soft real-time data aggregation application similar to [19], the message TTL is set to 5 hours and the message size is 512 Kbits.

Messages are generated randomly among all taxis and the network message generation interval is set from infrequent generation 100 second/msg to very frequent generation 10 second/msg, to see how these schemes react to the increase of network traffic load. Message source (a vehicle) and destination (a region) are randomly chosen. We set the simulation time to 24 hours with the first 6 hours as warm-up period. All metrics are calculated as the average value.

C. Message Delivery Performance

For comparison, five performance metrics, including the delivery ratio, overhead ratio, average latency, average hop count and average buffering time, are discussed. Because the difference of the geocast performance is dominated by the geographic forwarding process (i.e., from the source to the target region in a unicast fashion), the above performance metrics are all calculated for this phase. Once a message

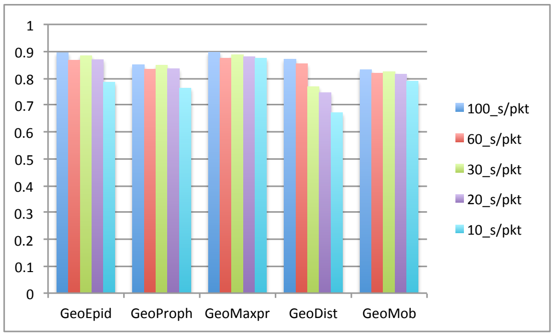


Fig. 6: Delivery ratio.

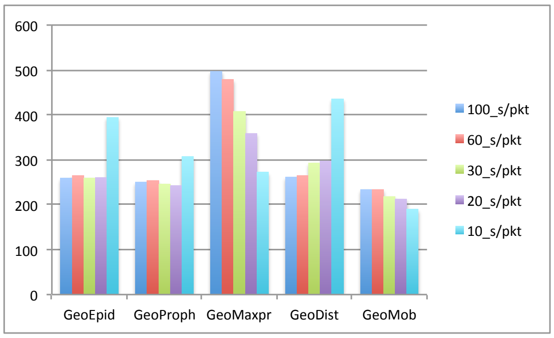


Fig. 7: Overhead ratio.

reaches the target region, we assume the same flooding scheme will be applied to all schemes. So the flooding performance should be the same for all schemes.

Figure 6 depicts the delivery ratio of the six schemes. Both Epidemic and MaxProp achieve very high delivery ratio, followed by our scheme and Prophet which are slightly lower, around 0.08% on average. We can see that the contact-based schemes are very effective in successfully delivering messages at the cost of extra communication and computation overhead as mentioned. On the other hand, our scheme is much less costly, and can achieve a similar performance. Distance-based geocast performs well with a low network traffic load. However, the delivery ratio drops significantly as the traffic load increases. As a comparison, our scheme and GeoMaxProp stay quite stable, or with a tiny drop, i.e., about 0.05%. This is achieved by the effectiveness of the routing decision and buffer management. Messages with high potential to be delivered always get transmitted first.

The overhead ratio of the schemes is shown in Fig. 7. It is defined as the ratio of the total number of message transmissions to the number of transmissions for those successfully delivered messages. A message can be removed for three reasons: 1) its TTL expires; 2) it is removed by the routing decision intentionally; 3) it is removed if the vehicle received an acknowledgement notifying that a copy of the message was successfully delivered to the destination.

Our scheme achieves the lowest overhead ratio under all network traffic load settings. This is because for each message, we strictly force all its transmissions to happen only on the optimal path determined by the macroscopic path selection process, as described in Section IV. Lots of message copies

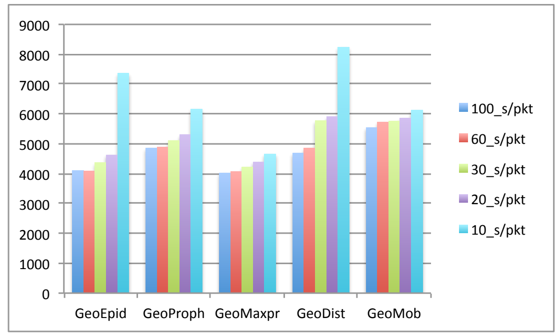


Fig. 8: Average latency (s).

are removed by the scheme intentionally to save storage and bandwidth resources. It is interesting to observe that when the traffic load is low, the overhead ratio of GeoEpidemic is even lower than that of GeoMaxProp. This is because with a very active buffer management, GeoMaxProp frequently removes messages from the buffer, creating more transmission chances for the saved messages going out and the new messages coming in. The buffer management can be efficient when the traffic load is high as the overhead is much lower than GeoEpidemic, but can also introduce extra overhead in a light traffic situation. The performance results show that our buffer management is just suitable.

For the overhead ratio trend with the increasing traffic load, we notice that many schemes have a sharp rise when the traffic load reaches the highest, showing that they cannot scale properly with a large traffic load. However, a very interesting and desirable observation is that, with the increase of the traffic load, our scheme and GeoMaxProp exhibit a decreasing trend on the overhead ratio. This can be explained by the effective buffer management implemented in these two schemes. In both buffer management schemes, messages in the transmission queue are sorted according to the potential of successful delivery, e.g., the likelihood; thus when a contact happens, the transmission opportunity is better utilized to help those messages with a higher delivery potential. Recall that the overhead ratio is calculated as the ratio of the number of total transmissions to the number of useful transmissions. According to the results, the increase of the useful transmissions surpasses the increase of the total transmission to some extent. In GeoDist, there is also a buffer management scheme based on the distance “improvement” as described earlier, but it does not achieve the scalability as ours does, implying that only the distance information may not be sufficient for buffer management.

Another important metric is the average latency, which is the average end-to-end delay of the delivered messages, shown in Fig. 8. Our algorithm shows a relatively higher latency, i.e., on average $1,200$ s, higher than the best one GeoMaxProp. The reason is that, in our scheme, messages are propagated following the optimal paths. Thus it may miss some shortcuts, with either a shorter geometry distance or fewer vehicle hops to the destination. Besides, with a low node density, the mobility pattern exhibited is not as strong

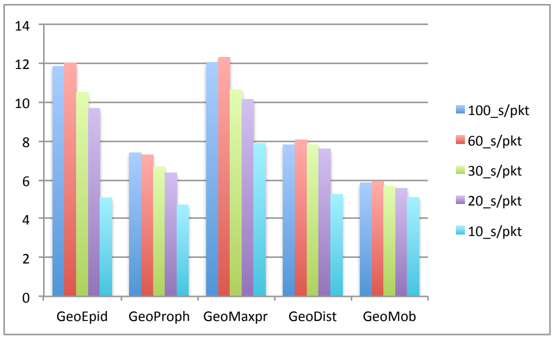


Fig. 9: Average hop count.

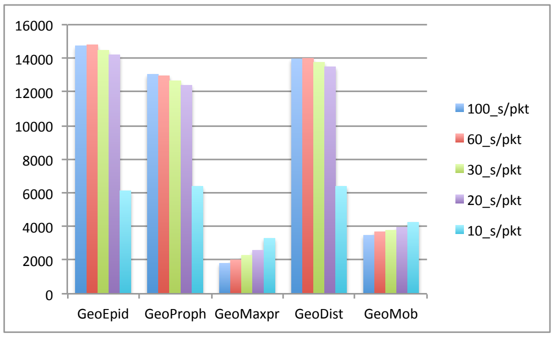


Fig. 10: Average buffering time (s).

as extracted from the trace with more than $4,000$ vehicles, leading to the reduction of performance. However, our scheme still scales better than the other schemes. With the increase of traffic load, GeoEpidemic, GeoProphet and GeoDist all experience a dramatic rise in average latency while our scheme always keeps small increases.

Figure 9 shows the average hop count of the delivered messages, which reflects the system overhead and complexity from another perspective. Apparently, our scheme has hop count around 5, which is much smaller than all the other schemes. Nearly for all schemes, the hop count decreases with the increase of the network traffic load. This is because, with more messages swarm in, the transmission chance for each message during a contact is getting smaller, leading to the reduction of transmission number and the hop count per message, including the delivered ones.

Figure 10 depicts the average buffering time, i.e., how long each message stays in the system, including the messages successfully delivered, intentionally dropped and expired. Both our scheme and GeoMaxProp achieve a very low average buffer time. Considering the high delivery ratio obtained, it implies that our scheme has very efficient buffer usage. The low buffering time is mainly achieved by the buffer management mechanism. GeoMaxProp drops messages which have too many hop counts or huge delivery cost, while our scheme drops messages when the carrying vehicles' future locations will not help the message propagation any more. Also, the delivery acknowledgement mechanism, which is adopted by both schemes, helps to drop redundant messages in the buffer.

VI. Conclusion

In this paper, an advanced mobility-aware geocast (GeoMob) scheme is proposed for large-scale urban VANETs. This new scheme has many new features compared with the existing geocast schemes: first, it is designed from a DTN point of view, enabling it to deal with the high mobility and transient connectivity in VANETs; second, different levels and aspects of vehicle mobility information is employed, making GeoMob very simple, scalable and communication and computation-effective; third, practical issues are well considered by introducing a real-world trace analysis, trace-driven simulation and efficient buffer management. Extensive performance comparisons with other protocols have shown the great advantages mentioned above.

References

[1] Y. Ko and N. Vaidya, "Geotora: A protocol for geocasting in mobile ad hoc networks," in *Proc. IEEE ICNP'00*, pp. 240–250, 2000.
[2] W. Liao, Y. Tseng, K. Lo, and J. Sheu, "Geogrid: A geocasting protocol for mobile ad hoc networks based on grid," *J. Internet Tech.*, 1(2):23–32, 2000.
[3] Y. Ko and N. Vaidya, "Flooding-based geocasting protocols for mobile ad hoc networks," *Mobile Networks and Applications*, 7(6):471–480, 2002.
[4] T. Imieliński and J. Navas, "GPS-based geographic addressing, routing, and resource discovery," *Communications of the ACM*, 42(4):86–92, 1999.
[5] T. Spyropoulos, K. Psounis, and C. Raghavendra, "Efficient routing in intermittently connected mobile networks: the multiple-copy case," *IEEE/ACM Trans. on Networking,* , 16(1):77–90, 2008.
[6] A. Lindgren, A. Doria, and O. Schelén, "Probabilistic routing in intermittently connected networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, 7(3):19–20, 2003.
[7] J. Burgess, B. Gallagher, D. Jensen, and B. Levine, "Maxprop: Routing for vehicle-based disruption-tolerant networks," in *Proc. IEEE INFOCOM'06*, pp. 1–11, 2006.
[8] G. Finn, "Routing and addressing problems in large metropolitan-scale internetworks," *DTIC Document*, 1987.
[9] I. Stojmenovic, A. Ruhil, and D. Lobiyal, "Voronoi diagram and convex hull based geocasting and routing in wireless networks," *Wireless communications and mobile computing*, 6(2):247–258, 2006.
[10] I. Leontiadis and C. Mascolo, "Geopps: Geographical opportunistic routing for vehicular networks," in *Proc. IEEE WoWMoM'07*, pp. 1–6, 2007.
[11] P. Cheng, K. Lee, M. Geria and J. Härri, "GeoDTN+Nav: geographic DTN routing with navigator prediction for urban vehicular environments," *Springer Mobile Networks and Applications*, 15(1):61-82, 2010.
[12] A. Vahdat, D. Becker and et al, "Epidemic routing for partially connected ad hoc networks," *Technical report CS-200006, Duke University*, 2000.
[13] H. Zhu, S. Chang, M. Li, K. Naik and S. Shen, "Exploiting temporal dependency for opportunistic forwarding in urban vehicular networks," in *Proc. IEEE INFOCOM'11*, pp. 2192–2200, 2011.
[14] H. Zhu, M. Dong, S. Chang, Y. Zhu, M. Li and X. Shen, "ZOOM: Scaling the mobility for fast opportunistic forwarding in vehicular networks," in *Proc. IEEE INFOCOM'13*, pp. 2832–2840, 2013.
[15] R. Hall, "An improved geocast for mobile ad hoc networks," *IEEE Trans. on Mobile Computing*, 10(2):254–266, 2011.
[16] A. Keränen, J. Ott, and T. Kärkkäinen, "The ONE simulator for DTN protocol evaluation," in *Proc. ICST SIMUtools'09*, pp. 55, 2009.
[17] M. Amadeo, C. Campolo, and A. Molinaro, "Enhancing ieee 802.11 p/wave to provide infotainment applications in vanets," *Ad Hoc Networks*, 10(2):253–269, 2012.
[18] H. Hartenstein and K. Laberteaux, "A tutorial survey on vehicular ad hoc networks," *IEEE Communications Magazine*, 46(6):164–171, 2008.
[19] C. Lochert, B. Scheuermann, and M. Mauve, "Probabilistic aggregation for data dissemination in vanets," in *Proc. ACM VANET'07*, pp. 1–8, 2007.