Advances in Intelligent Transportation Systems in Smart Cities - Research Article

International Journal of
Distributed
Sensor Networks

International Journal of Distributed Sensor Networks
2020, Vol. 16(5)
© The Author(s) 2020
DOI: 10.1177/1550147720907829
journals.sagepub.com/home/dsn
SAGE

Timely directional data delivery to multiple destinations through relay population control in vehicular ad hoc network

Seona Lee, Sang-Ho Lee and HyungJune Lee

Abstract

In this article, we consider a directional data forwarding problem to multiple destinations under distinct deadline constraints in vehicular ad hoc networks. We present a simple yet effective data forwarding algorithm based on only vehicle-to-vehicle communications in infrastructure-less and map-less environments. Our algorithm consists of two phases: *relay selection* and *proliferation*. We design a relay selection algorithm that encourages a shared ride for data delivery toward a certain common intermediate point from time to time for forwarding efficiency. It chooses a strong next relay candidate among nearby connected vehicles by considering their current position, velocity, and also the current progress toward the destination. In case that one of the progress lagging indicators becomes signaled, the number of vehicle relays increases under control depending on the degree of deterioration during a packet replication procedure called *proliferation*. Embedding two essential parts in designing a timely data forwarding scheme validates its accurate on-time data delivery performance and forwarding efficiency in network overhead based on real-world data-driven experiments.

Keywords

Deadline-aware routing, directional delivery, multicast, vehicle-to-vehicle communications

Date received: 24 June 2019; accepted: 29 January 2020

Handling Editor: Stephan Reiff-Marganiec

Introduction

With the emergence and rapid proliferation of autonomous driving and intelligent vehicles, the distributed usage of vehicle-to-vehicle (V2V) communication is known to be a fundamental way of exchanging information in vehicular ad hoc networks (VANETs). In particular, vehicle safety technology aims to ensure the safety of vehicles, passengers, pedestrians, and other vehicular facilities based on V2V networking. Also, developing an intelligent transportation system (ITS) that can reflect on-site road status or traffic information directly from vehicles or from a network of vehicles is a crucial part of building a smart traffic congestion control system.

To provide on-site road, traffic, and accident information to local police stations, emergency centers, fire stations or highway interchange/entrance/exit for timely emergency response, or on-the-road services such as drive-through cafe/restaurant, designing a viable V2V data forwarding mechanism to be responsive to ever-changing road situations is a requirement. At the same

Department of Computer Science and Engineering, Ewha Womans University, Seoul, South Korea

Corresponding author:
HyungJune Lee, Department of Computer Science and Engineering, Ewha Womans University, Seoul 03760, South Korea.
Email: hyungjune.lee@ewha.ac.kr

time, meeting timely delivery within a designated deadline with low network overhead is another challenge toward a well-balanced trade-off between time and efficiency.

There exist many real-world situations where locally detected road information or on-the-road services such as traffic density or preceding service request needs to be delivered to multiple destinations with distinct deadlines. Sometimes, even a single same event may need to be treated with different priorities depending on the type of multiple recipients: for example, in real-time or within a few seconds toward nearby vehicles or highway entrances and within some more relaxed time of a few minutes or several hours toward state-level traffic control centers. The time-sensitive data delivery to multiple destinations with each distinct deadline has been one of the challenging problems in VANET. It is because the delivery mechanism should be designed to neither over-waste nor under-waste network resources for achieving reliable timely delivery performance.

The problem of data delivery in VANET has been investigated mostly in infrastructure-assisted, map-based, and non-delay-tolerant environments.[1,2] Using pre-installed infrastructure nodes such as road-side units (RSUs) or access points as temporary stationary relays, routing paths can be established on top of a relatively stable underlying network.[3,4] Furthermore, explicit identification of junctions or anchors extracted from pre-loaded maps allows geographic routing approaches to overcome the local maximum problem due to its locality and traffic-aware street topology information.[5–8] However, infrastructure-less and map-less approaches with specific delivery time requirement have not been well studied, and thus need to be investigated for their even more practical applications.

To design an efficient data delivery mechanism toward multiple destinations without using any infrastructure or map information, it is desirable to effectively leverage an allowed delivery time for each destination. Furthermore, encouraging a shared ride for data toward a certain common intermediate point from time to time would be preferred for data forwarding efficiency. For instance, a courier company conveys numerous packages together with similar delivery due to a certain common hub and then distributes them respectively to each different destination from there. By effectively arranging shared and separate delivery methods depending on given delivery times, it is possible to devise a more efficient feasible delivery mechanism.

To handle the deadline-aware delivery progress, some lagging indicators considering the remaining deadline, the relative distance from the current vehicle to each destination, and so on should be defined. If a vehicle checks if one of the lagging indicators turns active, some more aggressive delivery action (e.g. through packet replication) needs to be performed. Otherwise, it can keep the current delivery method or revert back to the original frugal delivery method.

In this article, we propose an agile yet efficient directional data forwarding scheme with neither infrastructure nor digital map in VANET. Our data forwarding algorithm consists of two phases: (1) relay selection and (2) packet replication. In the relay selection phase, a single-vehicle relay finds out a strong next relay candidate among nearby connected vehicles, considering not only their current position, but also its derivatives (i.e. velocity), and also the current progress relative to a given delivery due in a greedy manner. In case that one of lagging indicators becomes signaled, the current vehicle relay decides to increase the number of next relays from one to a suitable pre-selected number. The number of relays to use is selected based on the degree of deterioration during a packet replication procedure called *proliferation*.

The advantages of our approach are twofold: lightweight yet effective relay selection criteria are selectively applied to designing a data forwarding scheme and validated in real-world datasets, and by adapting a data forwarding based on real-time network and delivery progress situation through controlling the increase of relay population, designing a compact yet timely reliable data forwarding is achieved.

Related work

The problem of directional data delivery in VANET has been investigated with three different perspectives: (1) infrastructure-assisted or infrastructure-less data delivery, (2) map-based or map-less data delivery, and (3) non-delay-tolerant or delay-tolerant data delivery.

First, data delivery schemes can be classified into two categories depending on whether to rely on static infrastructure nodes such as RSUs or access points. By embracing infrastructure devices as stationary nodes with different roles of source, relay, or destination, the underlying volatile data delivery problem in VANET due to highly dynamic topology of vehicles can be mitigated using both forms of communications, which are V2V and vehicle-to-infrastructure (V2I).[1,2] The optimal paths are obtained by estimating the delay of message delivery between static nodes,[9,10] by a series of sector-to-sector message delivery where a sector is governed by an RSU,[3] or by predicting a sequence of valid junctions to fixed infrastructure.[4] A recent study further considers the quality-of-service (QoS) metric based on considering connectivity, delivery ratio, and delay.[11]

As a new trend in the category of infrastructure-assisted data delivery in VANET, the software-defined network (SDN) concept has been integrated with the VANET context. Basically, SDN separates the control plane and the data plane to offer flexible network services. RSUs can gather the nodes' connectivity information to schedule data dissemination,[12] elect cluster heads,[11,13] or predict the mobility with the help of machine learning techniques.[14]

Second, most of the vehicular routing protocols in infrastructure-less environments have been developed under the name of geographic routing. Without using prior map information, greedy perimeter stateless routing (GPSR)[15] basically operates in a greedy-based routing and changes the mode to perimeter routing to recover from a local maximum, working in a well-distributed node environment. GPSR + AGF (advanced greedy forwarding)[16] improves its neighbor table considering the velocity, speed, and direction of vehicles, while spatially aware routing (SAR)[17] tries to overcome the local maximum by finding an alternative path with surrounding spatial awareness. This class of geographic routing usually has a limitation of working poorly in a relatively sparse vehicle environment. Instead of leaning on the geographic property, some research work has focused on handling collisions or hidden terminal problems by adapting the handshake mechanism or enhancing the medium access control (MAC).[18,19]

Under explicit identification of junctions or anchors from pre-loaded digital maps, this class of geographic routing protocol is applied. A classic geographic source routing (GSR) protocol[20] utilizes the locality of all the junctions from the source to its destination to apply the Dijkstra shortest path algorithm. Spatial and traffic-aware routing (STAR),[6] grid-based predictive geographical routing (GPGR),[5] and greedy traffic-aware routing (GyTAR)[7] do both locality and traffic-aware street topology information, to overcome the local maximum problem. However, for regions where their prior map information is not given or outdated, these advanced routing protocols may work poorly in practice.

Third, most of the aforementioned routing protocols work for non-delay-tolerant networks (DTNs) where data packets are transmitted to the destination as soon as possible at their own best effort. These non-delay-tolerant routing protocols do not utilize any margin time to find better connectivity at preceding times. On the other hand, there exist many classic delay-tolerant routing protocols such as epidemic routing[21] and *Spray-and-Wait* algorithm[22] known for their simple yet effective delivery performance. Recently, various research works such as adaptive carry-store forward (ACSF),[23] distance-aware routing with copy control (DARCC),[24] and geographic delay-tolerant network routing (GeoDTN+)[25] are conducted in the category of vehicular delay-tolerant networks (VDTNs). In VDTN, the main difference from the original DTN is that vehicles show some more characteristic patterns in daily traffic or so. It is more likely possible to track vehicles' frequently visited active areas and calculate the traffic-aware shortest path.[26]

More related to our work, delay-tolerant routing protocols such as scalable knowledge-based vehicular routing (SKVR),[27] vehicle-assisted data delivery (VADD,[10] and GeOpps[28] use the carry-and-forward technique that stores and carries data packets if proper vehicles cannot be found due to link volatility or highly dynamic topology change in VADD.

Although these prior works have contributed to the VANET routing research, devising some more practically feasible research work with infrastructure-less, map-less, and specific delay-sensitive data delivery requirements is necessary as an essential underlying vehicular communication method in the era of autonomous driving. This article presents a novel compact vehicular routing protocol complying with a delivery deadline constraint without using any infrastructure and map information.

System overview

In a scenario that road-side information including traffic situations or on-the-road services needs to be delivered to multiple different destinations such as nearby highway interchanges, road service centers, or emergency call centers, each respective delivery deadline should be considered upon deciding a forwarding path, depending on destination type, physical distance, and severity degree.

We consider the problem of data delivery to multiple destinations within time constraints in VANET. We seek a lightweight networking approach based on directional packet delivery, not relying on any given map or additional hardware tweak such as directional antenna installation. We use only V2V communication in a distributed VANET. The objective of this article is to design a best-effort data delivery mechanism that effectively delivers multiple destinations based on salient vehicle selection as intermediate relays by their distinct time deadlines while incurring low routing overhead.

We assume that each vehicle is equipped with a GPS device to locate itself and collects its past location trajectory in a buffer to calculate the average velocity of the vehicle itself during a past time window. It is also assumed that a vehicle communicates with another vehicle using a single forwarding packet with a wireless interface (e.g. 802.11p). Although the maximum frame body size for 802.11p within a packet is known to be 2304 bytes, many studies limit the packet size to 1000 bytes, concerning throughput and other issues.[29,30] We also assume that the data size for one forwarding packet is less than 1000 bytes where the required information can be contained in a single data packet.

To perform on-time data delivery from a source to multiple destinations in a distributed VANET environment, finding a series of reliable vehicle relay selection

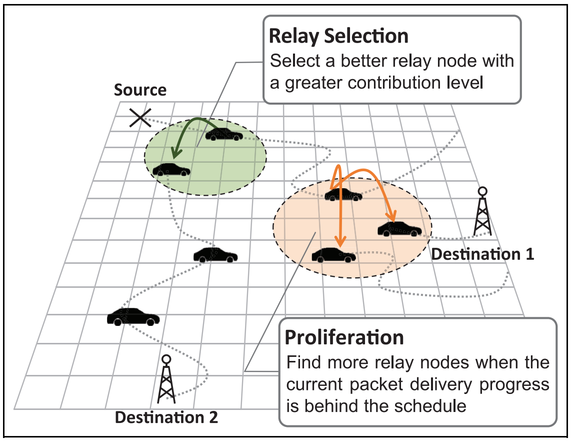
Relay Selection
Select a better relay node with a greater contribution level
Source
Destination 1
Destination 2
Proliferation
Find more relay nodes when the current packet delivery progress is behind the schedule

Figure 1. System overview with two phases of vehicle relay selection and proliferation depending on the overall timely delivery progress.

is necessary. Since the movement of nearby vehicles as relay candidates tends to be very dynamic, the selection of whom the current vehicle should deliver data toward each destination very challenging. On a regular basis, the selection decision should be made among multiple relay candidates based on a contribution level related to *where* and *how long* a vehicle candidate will successfully be carrying data toward a designated destination. Our proposed scheme consists of two main phases: (1) *relay selection* and (2) *proliferation*, as shown in Figure 1.

During the *relay selection* phase, in order to make on-road vehicle resource minimally engaged for efficient delivery, it would be necessary to cover with a single-vehicle relay to deliver data toward many destinations in a similar direction. When data get much closer toward those destinations and need to be split toward each different destination at some point, multiple vehicle relays need to be in charge of separate delivery to each destination. The relay selection procedure is described in section "Relay selection."

In the *proliferation* phase, we want to *proliferate* data through packet replication to make aggressive progress for being behind the delivery schedule. If the current vehicle relay finds out that its next data relay to one single-vehicle is not enough to reach its planned destinations on time, it chooses multiple vehicle relays and forwards the data to them. The packet replication procedure is described in section "Proliferation."

Relay selection

In this section, we present a relay selection algorithm that finds out an effective vehicle relay node that can contribute to carrying data for a while until a next relay takes over. Upon selecting one out of multiple vehicle candidates as the next relay node, we have to prioritize them by quantifying how much each candidate can contribute to carrying the data closer toward a given destination.

To quantify how a vehicle can be used as a message ferry by moving closer toward a given destination, we use the average velocity of recent movements of a vehicle relay candidate during the latest few minutes (e.g. last 5 min). We decompose the vector of velocity into the direction to the destination. As illustrated in Figure 2, the magnitude of the average velocity vector $\vec{v}$ to the direction of the destination is given by $|\vec{v}|\cos\theta$, where θ denotes the angle between the velocity vector and the direction of the destination. Assuming that the average movement vector keeps valid until the remaining delivery deadline t_{rest}, we can calculate the estimated travel distance toward the destination if a vehicle relay candidate is selected as follows

$$\hat{d} = |\vec{v}|\cos\theta \cdot t_{rest} \tag{1}$$

We define a *contribution level* ϕ measure of how the effective travel distance $\hat{d}$ toward a destination is estimated compared to the rest distance d_{rest}, from the current position to the destination as follows

$$\phi = \frac{\hat{d}}{d_{rest}} = \frac{|\vec{v}|\cos\theta \cdot t_{rest}}{d_{rest}} = \frac{|\vec{v}|}{|\vec{v}_{target}|} \cdot \cos\theta \tag{2}$$

where $|\vec{v}_{target}|$ denotes an ideal speed, which is the shortest distance d_{rest} toward the destination divided by the remaining delivery time t_{rest}. We want the estimated travel distance $\hat{d}$ enough to be larger than d_{rest}, that is, $\phi>\phi_c$ (where $\phi_c \geqslant 0$), implying that the vehicle can more likely arrive at the destination within its remaining deadline based on the recent movement history.

We calculate each contribution level for all possible nearby vehicle relay candidates from the current relay on a regular basis (e.g. every 3 min) and use the measure as information deliverability. We prioritize them based on the order of measure for selecting its next relay vehicle.

Single destination

In the case of data delivery to a single destination, the current vehicle that holds the packet needs to select a single next relay vehicle that can move it closer toward the destination, compared to itself. On a regular basis, the current relay node considers all of the nearby vehicles as relay candidates for the next round and then chooses one of them as the next relay node. To find adjacent vehicle nodes, the current relay pings surrounding vehicles near its vicinity by sending a HELLO packet. Any nearby vehicle that has received the ping message replies with its vehicle ID and past location trajectory information.

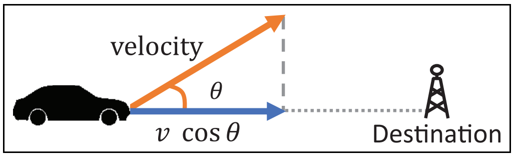

Figure 2. Evaluating nearby relay candidates by calculating the effective travel distance with the average velocity toward the direction of a given destination.

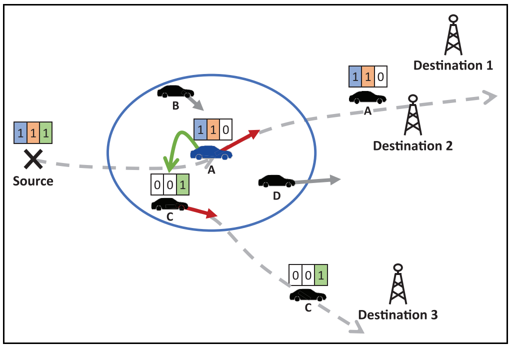

Figure 3. Example of relay selection on multiple destinations.

Table 1. Example of contribution values table.

	Destination 1	Destination 2	Destination 3
Vehicle A	1.2	1.5	0.3
Vehicle B	–0.2	1.7	1.4
Vehicle C	0.3	0.2	4.3
Vehicle D	1.1	1.2	1.1

When the current relay vehicle receives all the information from its neighboring vehicles at that time, it calculates the contribution level measure ϕ over all responded vehicles including itself and selects a vehicle with the highest value as the next relay vehicle and relays the packet to it. If it does not receive any response for its previous ping message, it automatically becomes the next relay vehicle and holds the packet by the next selection period.

Multiple destinations

If the initial vehicle or the current relay vehicle is in charge of delivering information toward multiple destinations, we want to make fewer relay vehicles engaged with the relaying task. We aim to save radio energy consumption for transmission and reception in each vehicle side and also reduce network traffic and overhead in the entire network side. If one relay vehicle moves toward several destinations similarly aligned with a certain direction range, it can carry the information up to some point as a shared relay. After that point, the original single data path may need to be split, and the data are relayed from the current vehicle to multiple distinct vehicles. These vehicles are responsible for their respective data delivery to each different destination, as illustrated in Figure 3.

Based on the received information of adjacent vehicles and their trajectory information, the current relay vehicle calculates the contribution value for each vehicle candidate and each destination as shown in Table 1. Using the table of contribution values calculated across vehicles and destinations, we find a set of vehicles that can effectively cover all the destinations while having higher contribution values across them relative to the number of selected vehicles. In case that the number of selected vehicle relays for the next round is larger than 1, it is time to replicate the currently carrying packet and relay it to the selected multiple vehicles.

For example, assuming that vehicle A is the current relay node and it finds the adjacent vehicles B, C, and D, vehicle A traverses all possible vehicle set combinations that can cover all the given destinations by satisfying their contribution values larger than or equal to ϕ_c. As shown in Table 1, vehicles A and B can cover all three destinations, while vehicle D alone can also cover all of them. To choose the most effective set of relay vehicles, we calculate the normalized sum of the highest contribution values over destinations by the number of vehicles as an efficiency measure. The case of vehicles A and B has the efficiency measure of 2.15 $(= (1.2 + 1.7 + 1.4)/2)$ based on the contribution values of 1.2 (by vehicle A), 1.7 (by vehicle B), and 1.4 (by vehicle B), while that of vehicle D has the value of 3.4 $(= (1.1 + 1.2 + 1.1)/1)$. In this example, the set of vehicles A and C ends up with the highest efficiency of 3.5, meaning that vehicle A for destinations 1 and 2 and vehicle C for destination 3 will be used as next relay vehicles. It should be noted that this problem of covering multiple destinations by a limited set of vehicles is a variant of the classic set cover problem where various efficient heuristic algorithms can be applied to solve.[31,32]

In case we cannot find a complete set of vehicles that can cover all the destinations, we choose every single vehicle that can cover each destination with the highest contribution value. After solving the set cover problem with the highest efficiency, the current relay vehicle relays the information to the selected next relay vehicles, which are dedicated to data delivery to certain destinations. The detailed relay selection procedure is described in Algorithm 1.

Proliferation

Even if any advanced design of relay selection is embedded in on-time data delivery mechanisms, it is

Algorithm 1. Relay selection algorithm.

```
Input: $\vec{v}$, destination set $D$, source $src$, remaining time $t_{rest}$
Output: new relay vehicle $node_n$ corresponding to $dst_n$
// Perform relay selection in a regular basis
if relay selection timer expired then
    $relaySelectionFlag = true$;
end if
// Check if proliferation is needed
if proliferation timer expired then
    Invoke the function proliferation($t_{rest}$, $D$, $src$);
end if
// Perform relay selection
if $relaySelectionFlag == true$ then
    Request adjacent nodes' trajectory information;
    // Contribution values table and coverage calculation
    for each adjacent node $node_n$ do
        $coverage_n = \emptyset$;
        for each destination $dst_i \in D$ do
            $\phi_{n,i} = \hat{d}_{n,i}/d_{rest_{n,i}}$;
            if $\phi_{n,i} > \phi_c$ then
                $coverage_n = coverage_n \bigcup dst_i$;
            end if
        end for
    end for
    // Find the most efficient vehicle set
    for every vehicle set combination $S$ do
        if $\bigcup_{n \in S} coverage_n \neq D$ then
            continue;
        end if
        if $proliferationFlag == true$ then
            Invoke the function proliferationCheck($D$, $S$,
            $proliferationNum$, $proliferationFlag$);
            // If $S$ does not satisfy the proliferation criteria, skip
            if $complyFlag == false$ then
                continue;
            end if
        end if
        // In case that $S$ meets all conditions, calculate $efficiency$
        $efficiency = \sum_{i \in D} \max \phi_{n,i}/|S|$;
        Record the vehicle set with the maximum $efficiency$;
    end for
    if no vehicle set for full destination coverage then
        Choose each vehicle with the highest $\phi_{n,i}$ for each $dst_i$;
    end if
end if
Selected new vehicles play a role in carrying information
toward each destination;
```

still very challenging to ensure its timely data delivery using only a single relay dedicated to one destination. Since real-world vehicle traces usually include erratic mobility patterns, even carefully selected relays do not often meet the original expectation, for example, arriving at the destination beyond the given deadline or wandering around somewhere else.

To tackle the problem, we borrow the concept of *proliferation* from biology in designing a packet replication scheme in the VANET context. If the current progress status expects to be pessimistic for on-time data delivery to a certain destination, we want to make rapid reproduction of information by outnumbering the relay vehicles. Through the *proliferation* process, we anticipate a high probability of delivering information at its intended destination within its given deadline.

However, if the proliferation process performs too aggressively, the overall network should experience an unnecessarily large amount of packet overhead, while wasting too much energy consumption for packet transmission and reception at all the involved vehicles. To effectively control the degree of packet replication, it is important to set a proper criterion of *when to turn on* the proliferation mode and *how aggressively* it should perform.

To derive the criterion, we quantify the overall vehicle relaying progress toward each destination. If the current single relay vehicle cannot afford to perform its original mission expected by its previous relay vehicle, the relay should determine how many relay vehicles are necessary to take over the mission by quickly recovering the lagging delivery situation.

As illustrated in Figure 4, we use the original distance d and the remaining delivery due $t_{deadline}$ from a source to a destination, and the remaining distance d_{rest} and time t_{rest} from the current vehicle position to the destination. Using this notation, we define a reference velocity $\vec{v}_{ref}(=\vec{d}/t_{deadline})$ as the initial velocity at source, and the target velocity $\vec{v}_{target}(=\vec{d}_{rest}/t_{rest})$ at the current relay vehicle should reach in remaining of time. On a regular basis, a relay node checks its current progress by comparing the target speed $|\vec{v}_{target}|$ with its initial reference speed $|\vec{v}_{ref}|$. The condition that $|\vec{v}_{target}|$ is larger than $|\vec{v}_{ref}|$ shows that the current position toward the destination runs behind schedule. This means that the current and past relay selection has not been good enough to make progress.

In this situation, we launch the proliferation phase by calculating the necessary number of replication packets and then choosing suitable relay vehicles. We apply the concept of acceleration to catch up the progress lagging, that is, $|\vec{v}_{target}| - |\vec{v}_{ref}|$ over the next remaining time t_{rest}. We decide to replicate the current relay packet, proportionally to the above force as follows

$$[Number\ of\ replication] = \alpha \cdot \frac{(|\vec{v}_{target}| - |\vec{v}_{ref}|)}{t_{rest}} \tag{3}$$

where α is a constant scaling factor to control the replication degree.

In case that the required number of relays is larger than the total available relay vehicles that satisfies the contribution value threshold (i.e. $\phi > \phi_c$), only the available vehicles are selected as next relays. Once the number of replication is determined, the current relay vehicle looks through the contribution table as calculated in the relay selection procedure. Then, it finds the most effective vehicle set with a sufficient number of

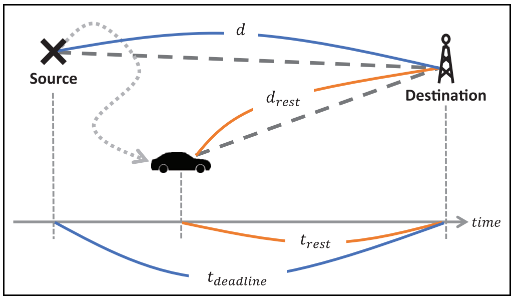

Figure 4. Estimating the number of replication by comparing original expectation and current progress.

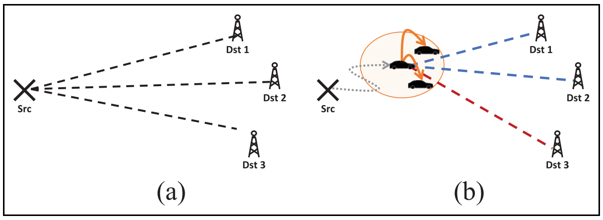

Figure 5. Data forwarding progress example for proliferation: (a) t = 0 and (b) t = 10.

vehicles that can cover all the given destinations. Both α and ϕ_c affect the degree of replication. For example, the lower ϕ_c is, the larger number of relay candidates is considered upon selection.

For instance, we consider an example case as shown in Figure 5 and Table 2. The current relay vehicle checks the progress of the current data forwarding for all destinations at $t = 10$. For destination 1, since $|\vec{v}_{target}|$ is less than $|\vec{v}_{ref}|$, we consider its data forwarding as well-paced. Regarding destination 2, $|\vec{v}_{target}|$ is larger than $|\vec{v}_{ref}|$, and the number of replication can be calculated as $\lfloor 20 \cdot (2.16 - 2)/30 \rfloor = \lfloor 0.1066 \rfloor = 0$ (where $\alpha = 20$). This means that any further vehicle relay is not needed beyond the current one for destination 2. For destination 3, on the other hand, the metric is calculated as $\lfloor 20 \cdot (2.5 - 1.5)/10 \rfloor = 2$. Therefore, the current relay vehicle chooses two more relays, meaning that a total of three vehicle relays will become in charge of destination 3 in the next round.

The detailed proliferation procedure is described in Algorithm 2.

Table 2. Example of proliferation degree decision.

	t = 0			t = 10		
	d	$t_{deadline}$	$\lvert\vec{v}_{ref}\rvert$	d_{rest}	t_{rest}	$\lvert\vec{v}_{target}\rvert$
Destination 1	40	20	2	15	10	1.5
Destination 2	80	40	2	65	30	2.16
Destination 3	30	20	1.5	25	10	2.5

Algorithm 2. Proliferation algorithm.

```
function proliferation(deadline $t_{rest}$, destination set $D$, source $src$)
// Calculate the number of replication for each destination
for each destination $dst_i \in D$ do
    $proliferationNum_i = (|\vec{v}_{target_i}| - |\vec{v}_{ref_i}|)/t_{rest_i} \times \alpha$;
end for
if $proliferationNum_i \geqslant 1$ then
    $proliferationFlag_i = true$;
end if
return $proliferationNum$, $proliferationFlag$
endFunction
function proliferationCheck(destination set $D$,
vehicle set $S$, $proliferationNum$, $proliferationFlag$)
// Check if the vehicle set satisfies the proliferation criteria
$complyFlag = true$
if there exists $dst_i$ with $proliferationFlag_i = true$ then
  if $|\{n|n \in S, dst_i \in coverage_n\}| < proliferationNum_i$ then
    $complyFlag = false$;
  end if
end if
return $complyFlag$
endFunction
```

Evaluation

We evaluate our proposed directional data forwarding algorithm based on real-world data-driven simulations. A taxi trace dataset, Shanghai urban vehicular network (SUVnet)[33] that was collected in Shanghai, China is used. Since this dataset does not have any V2V communication history, we simulate the direct packet communication among vehicles on top of vehicles' moving traces.

The simulated area is $20 \times 20\,\text{km}^2$ where 2147 vehicles were actively moved around, as shown in Figure 6. By setting one source location and multiple destinations in Figure 7, we validate network performance in terms of on-time packet delivery success rate and packet overhead, compared to several representative approaches as counterpart baseline.

Since the original real-trace dataset includes numerous noisy records such as duplication, unrealistic movements, and unnecessary short trips (less than 180 s), we have pre-processed and refined it. We have removed duplicated and erroneous trace records or filling out some missing records between consecutive records by linear interpolation for our experiments.

For our evaluation, 10-day data out of the whole 30-day dataset are used, that is, from 10 February 2007 to 19 February 2007. We have chosen the most crowded time periods at 11 a.m., 12 p.m., and 1 p.m. where each packet delivery launches. We run data forwarding algorithms including ours under the 30 different experiments and report the averaged measures. For

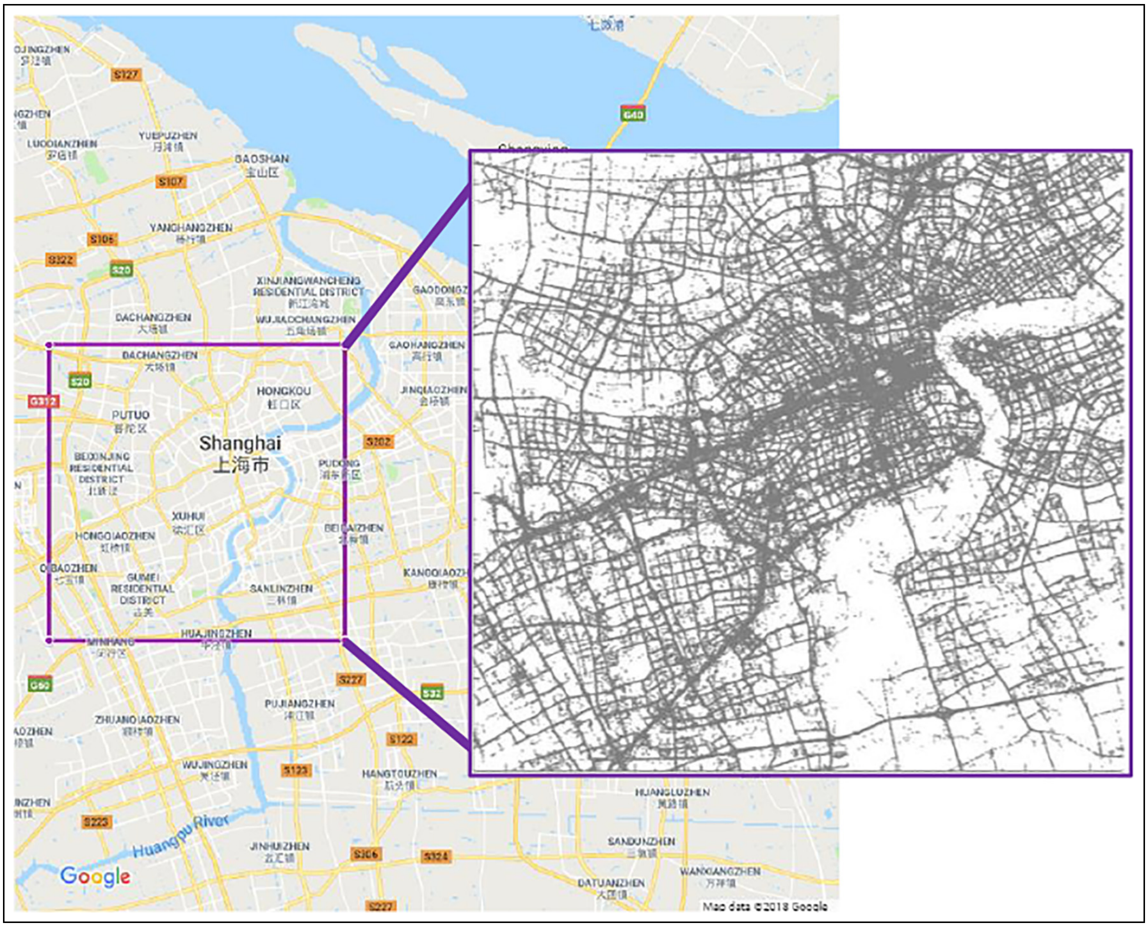

Figure 6. Real trace-driven simulation experiments over 20×20 km^2 in Shanghai, China where actual taxi traces are shown with its corresponding Google map of the area.

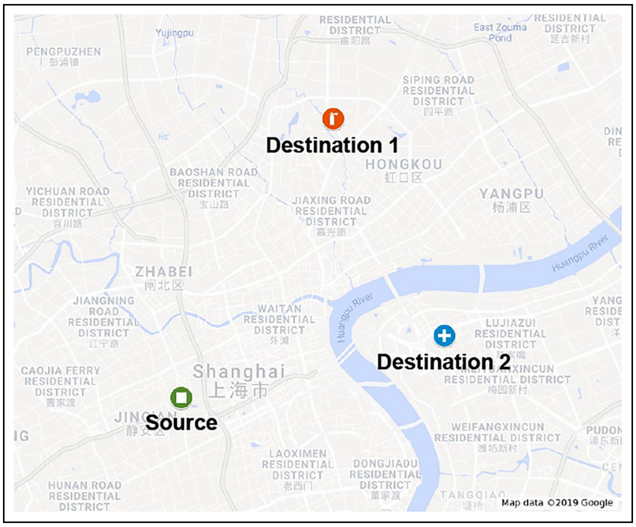

Figure 7. Data delivery scenario from one source to two different destinations in our experiments.

simulating direct vehicular wireless links, the transmission range of 300 m is used to be compatible with 802.11p.[34]

Our relay selection and proliferation algorithm runs every 3 min. The parameters of the contribution level threshold ϕ_c and the replication scaling factor α are tuned to be 1 and 20, respectively, unless otherwise noted. The environment and simulation parameters are summarized in Table 3.

Relay selection validation

We compare our relay selection algorithm against two other counterpart algorithms: a random relay selection and a greedy data forwarding scheme. The random relay selection algorithm does not consider any vehicle information such as its location, velocity, and deadline constraints and simply selects a single vehicle as to the next round relay among adjacent connected vehicles in a random fashion. The comparison between it and our algorithm may provide some interesting insight into how some essential information can improve timely data delivery in VANET. As another baseline, we let a relay selection algorithm prioritize nearby connected vehicles based on only their location. It chooses a

Table 3. Simulation environment and parameters.

Environment	Value
Simulation area	400 km^2
Number of active vehicles	2147
Communication range	300 m
Relay selection period	3 min
Proliferation period	3 min
Contribution level threshold, ϕ_c	0–1
Replication scaling factor, α	1–20

vehicle relay that has the closest to the destination in a greedy manner, which can be considered as a simplified version of GPSR algorithm.[15]

First, we examine network performance of each algorithm for two single destination cases, that is, toward destination 1 for test case 1 and toward destination 2 for test case 2, respectively, as shown in Figure 7. In these test cases, two destinations are located within a similar distance from the source, whereas destination 2 is located across the river, making packets only reachable to the end through bridges. We quantify a successful data delivery rate for delivery reliability and the number of unique relay vehicles used for delivery efficiency. We also measure transmission cost consisting of control overhead as the number of HELLO packets used in each case and data overhead as the number of data packet transmissions. We vary the packet delivery deadline from 30 min to 3 h, as shown in Figure 8.

As shown in Figure 8(a) and (d), all of the algorithms almost fail to deliver one packet to its destination within the deadline of 30 min since there may exist insufficient vehicles for data delivery from the source to each destination for both test cases 1 and 2. By relaxing the deadline constraint up to 3 h, our relay selection algorithm finds an effective way to improve on-time data delivery rate up to 73% for test case 1 and 53% for test case 2, respectively, by continuously finding one effective single relay vehicle at each time. The 20% performance gap between test cases 1 and 2 results from the fact that under the situation of using only one relay vehicle over time, the opportunity for the current relay vehicle to encounter a suitable vehicle toward the other end of the bridge toward destination 2 becomes practically too low.

The random relay selection scheme works very poorly over almost all experiments, while the closet relay selection scheme reaches up to 47% for test case 1

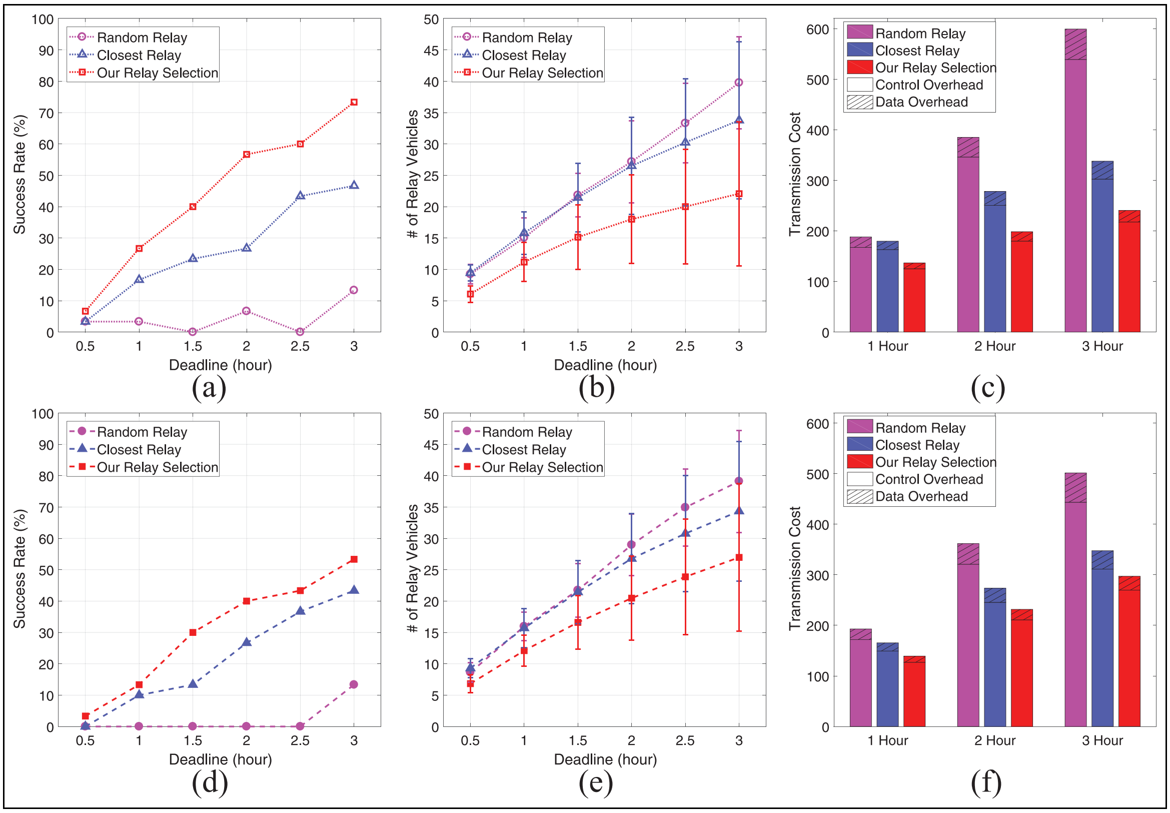

Figure 8. Network performance for single destination experiments toward destination 1 for test case 1 and destination 2 for test case 2. The standard deviation of the number of relay vehicles is shown for test cases 1 and 2: (a) on-time delivery rate on test case 1, (b) number of relay vehicles used on test case 1, (c) transmission cost on test case 1, (d) on-time delivery rate on test case 2, (e) number of relay vehicles used on test case 2, and (f) transmission cost on test case 2.

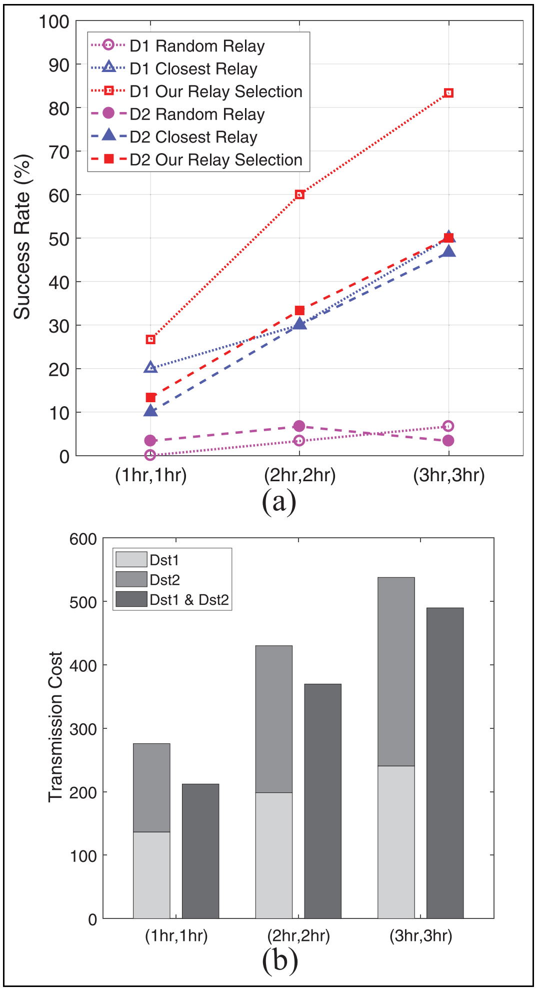

Figure 9. Concurrent data delivery performance under multiple destinations at the same time: (a) on-time delivery rate and (b) transmission cost for each separate delivery scenario versus concurrent delivery scenario.

and 43% for test case 2, respectively. The gap between our algorithm and the closest relay selection algorithm means that using not only location information, but also its first derivative and a progress indicator, for example, the remaining delivery time, helps to enhance on-time data delivery rate.

Regarding the number of relay vehicles and transmission cost, all of the algorithms spend more relay and communication resource due to more chances to relay as the deadline gets relaxed. Our relay selection algorithm incurs the lowest relay and communication resource thanks to a salient selection of representative effective relay vehicles, compared to others.

This result implies that although our relay selection algorithm fairly achieves both reliability and efficiency at the same time, using only one relay vehicle at a time is not sufficient for achieving practically high on-time delivery rate in practice. It requires an additional phase of make-up.

Second, we validate delivery performance on a concurrent data delivery scenario that initiates data forwarding from one source to multiple destinations under the same deadline constraints of 1, 2, and 3 h, respectively, as shown in Figure 9. As shown in Figure 9(a), our relay selection algorithm outperforms others while keeping almost similar on-time delivery performance compared to each respective delivery case in Figure 8(a) and (d).

We check how much our algorithm can share some common relay vehicle to different destinations. We quantify transmission cost of the concurrent data delivery to two destinations compared to the cumulative cost over each respective delivery case. The concurrent data forwarding based on our relay selection algorithm achieves up to 23% less transmission cost. This means that our algorithm selectively uses some effective shared rides that can cover multiple destinations as far as possible for efficient resource usage.

Proliferation validation

Now we evaluate our approach equipped with the proliferation algorithm that can compensate on-time delivery rate without wasting too much network overhead. We first focus on experiments on test case 2 where our proliferation-free relay selection algorithm only achieves up to 53%.

We validate on-time delivery rate and transmission cost with various proliferation parameter setting together with the original (proliferation-free) relay selection algorithm by varying the delivery deadline from 30 min up to 3 h in Figure 10.

As shown in Figure 10(a), the additional proliferation algorithm makes our relay selection algorithm significantly improve on-time delivery rate over up to 90% in the same environment. In particular, a certain parameter setting of ($\phi_c = 1, \alpha = 1$) in proliferation achieves a higher rate than the relay selection over all deadline cases with a moderate population. A parameter setting of ($\phi_c = 0, \alpha = 20$) achieves the highest delivery rate than other settings, considered as a high population.

To see how each parameter affects the performance, if the replication scaling factor α increases from 1 to 20, a larger number of relay vehicles with higher transmission cost (as shown in Figure 10(b)) contributes to data forwarding, further enhancing on-time delivery performance. If the contribution level threshold ϕ_c decreases from 1, 0.5 to 0, the on-time delivery rate becomes increased. This implies that under a lowered barrier for candidate selection, a more competitive

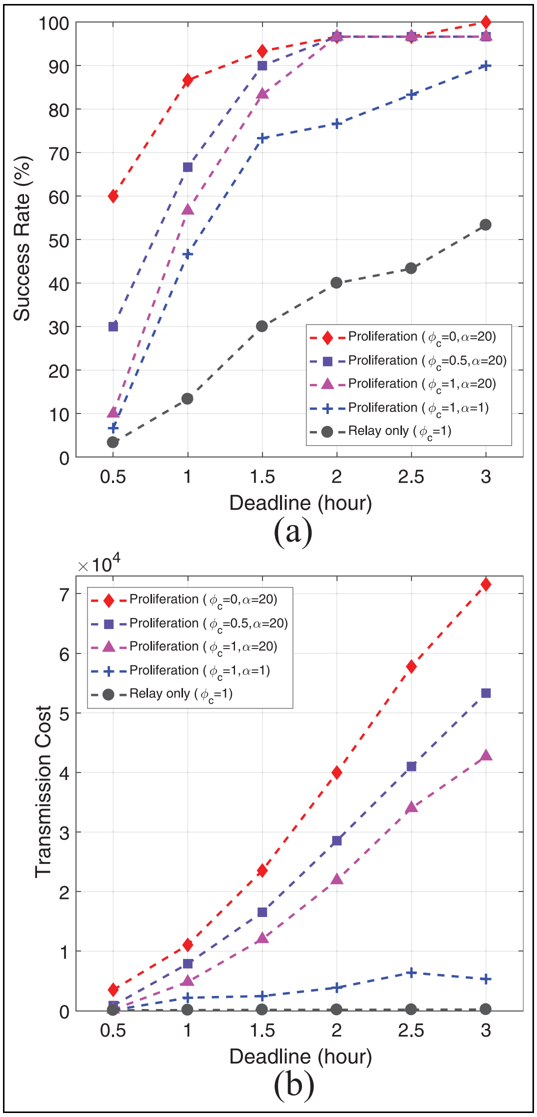

Figure 10. Single destination performance with proliferation with respect to deadline: (a) on-time delivery rate on test case 2 with proliferation and (b) transmission cost on test case 2 with proliferation.

vehicle set may be selected by contributing more significantly to the end.

Also, we take a closer look at performance dynamics between relay selection and proliferation in terms of distance to destination from the current relay and the number of relay vehicles used over time till the deadline as shown in Figure 11. A certain set of three representative test experiments are chosen under the deadline of 1.5 h: (1) both-success case in Figure 11(a) and (d), (2) only proliferation-success case in Figure 11(b) and (e), and (3) both-fail case in Figure 11(c) and (f).

In the both-success case, both algorithms successfully reach the destination within the deadline, showing the blue vertical line at that time as shown in Figure 11(a). The number of relay vehicles used in proliferation increases over time as shown in Figure 11(d). This can be considered as an optimistic case.

However, as a more general case, there exist more experiment cases where the single relay selection fails, but the one with proliferation succeeds. By leveraging more relay vehicles by turning on the proliferation mode as shown in Figure 11(e), successful on-time delivery is achieved through some moderate population increase in Figure 11(b). An interesting observation is that once a proliferation mode is enabled, and more relay vehicles become involved, the distance toward the destination becomes decreased. This means that selecting more competitive relay vehicles helps to contribute to its delivery success.

As a pessimistic case where both algorithms fail to deliver within the deadline, we observe that the distance continuously gets decreased as shown in Figure 11(c). This is because more relay vehicles are used over time in Figure 11(f), although its first arrival at destination is out of the deadline.

Finally, we validate network performance in the previous concurrent data delivery scenario where a packet leaves for two destinations at the same time. We compare our proliferation algorithm with a high population setting of ($\phi_c = 0$, $\alpha = 20$) together with other counterpart algorithms. They are random relay selection with probability p (where $p = 0.1$ is used) among all connected vehicles at a time and top K closest relay selection to destinations (where K is set to 2) in Figure 12. As a further comparison counterpart, we simulate a popular classic delay-tolerant routing, epidemic routing,[21] which forwards a packet whenever one node encounters another. The epidemic routing is expected to show the upper-bound performance in terms of delivery accuracy. These counterpart algorithms operate independently for each destination delivery.

Our proliferation algorithm achieves a high stable on-time delivery rate above 80% for both destinations when the deadline is 1 h and beyond as shown in Figure 12(a). Although the K closest relay selection holds the highest on-time delivery rate, it incurs a much higher transmission cost than ours with a factor of 10 as shown in Figure 12(b) and (c). Furthermore, the epidemic routing spends incomparably high transmission costs, whereas reaching 100% timely delivery. It is demonstrated that without delicate packet replication control, a delivery algorithm can waste a tremendous amount of transmission cost. This implies that our proliferation performs on-time data delivery toward

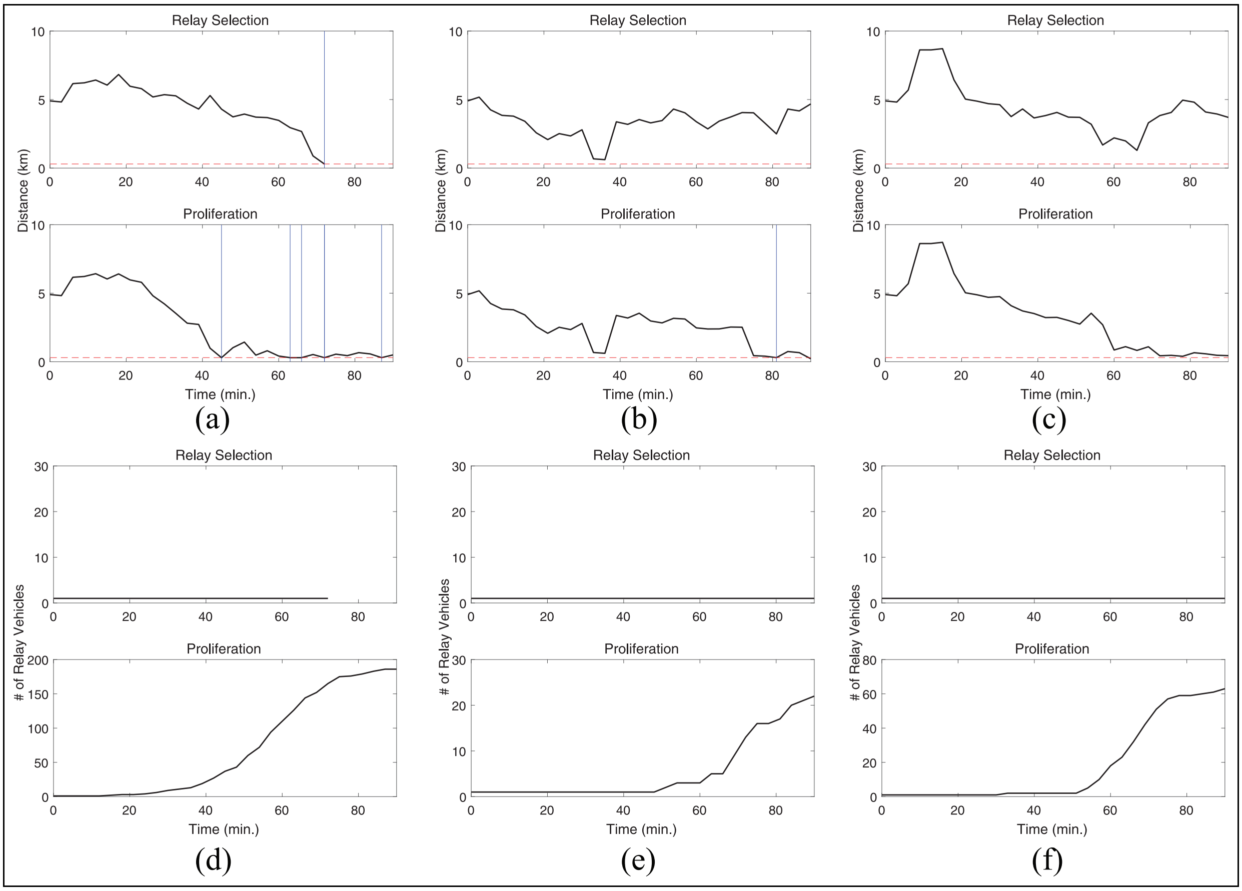

Figure 11. Relay selection and proliferation result comparison on test case 2 under the deadline of 1.5 h with $\phi_c = 1$ and $\alpha = 1$: (a) distance to destination in an optimistic test experiment, (b) distance to destination in a general test experiment, (c) distance to destination in a pessimistic test experiment, (d) number of relay vehicles used in an optimistic test experiment, (e) number of relay vehicles used in a general test experiment, and (f) number of relay vehicles used in a pessimistic test experiment.

multiple destinations at a practical balance between reliability and efficiency.

Conclusion

We have addressed the problem of directional data delivery to multiple destinations within distinct deadline based on relay selection and proliferation. We have presented a simple yet effective relay selection that encourages a shared ride for data delivery in parts on the way to each destination. By considering only essential vehicle information of position, velocity, and remaining packet delivery time, it continues to find an effective relay vehicle in a periodic manner and makes its best effort to arrive at destinations.

Since only a single relay vehicle is not sufficient for achieving practically stable on-time delivery performance in real-world situations, our approach turns on a proliferation mode that increases the number of relay vehicles. We dynamically control the population of relay vehicles depending on data delivery progress. By doing so, our data forwarding algorithm has validated its practically high performance in terms of both reliability and efficiency in on-time packet delivery in real-world data-driven simulation experiments.

Although our proposed algorithm shows its accuracy and efficiency in timely delivery, it still spends some significant network overhead in urgent situations. This limitation is mainly caused by the fact that even if some relay vehicles have already arrived at the destination, some other relay vehicles have no way to be aware of the delivery results while keeping replicating. As a future extension, we may consider a new design dimension such that each relay vehicle can self-control packet replication in a probabilistic way as the delivery deadline comes closer.

For other future works, we may devise a hybrid data forwarding approach that can use not only unconstrained vehicles but also some constrained vehicles such as buses that move along pre-determined routes. It may improve both packet delivery reliability and efficiency by arranging two different types of vehicles at each suitable time depending on the deadline budget.

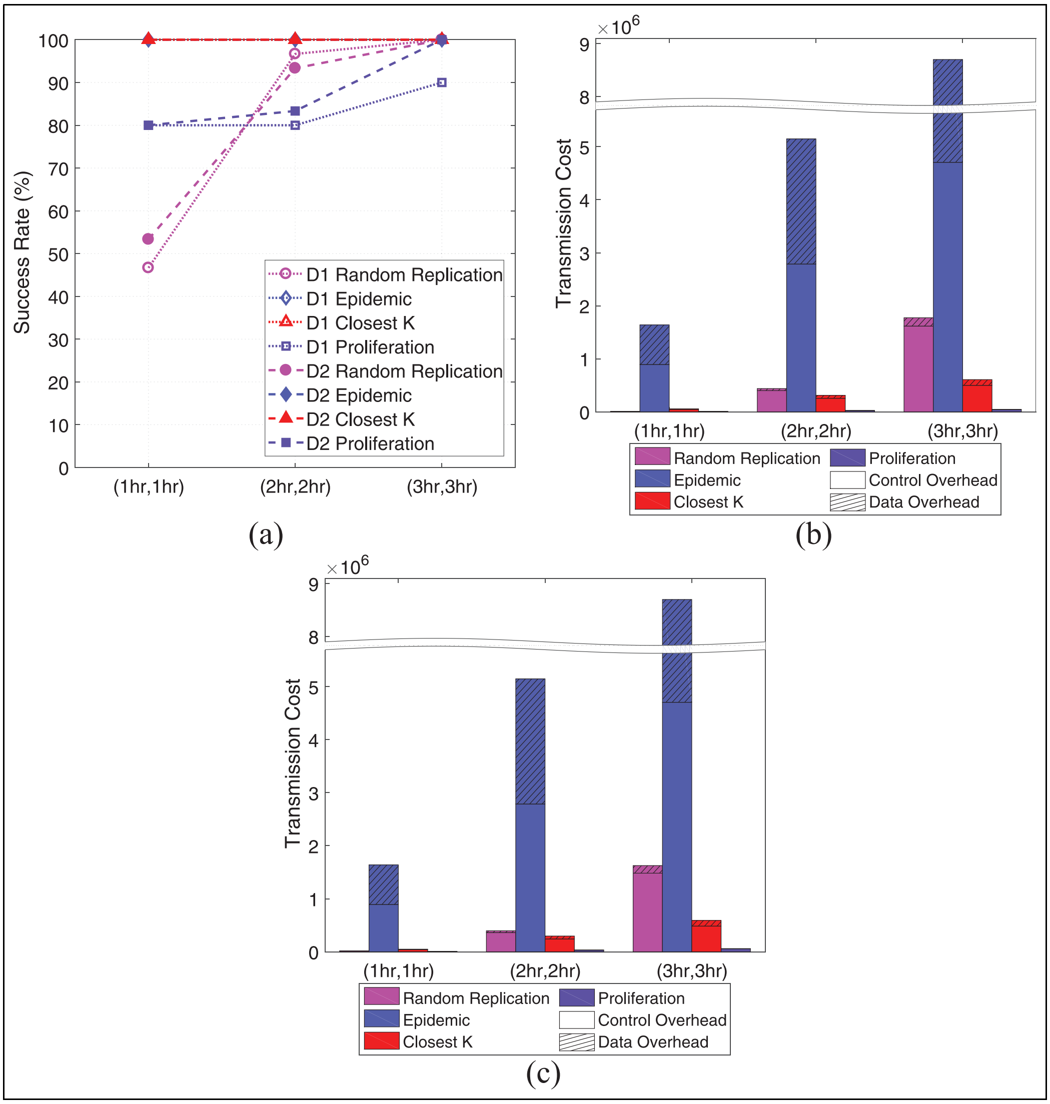

Figure 12. Concurrent data delivery performance to multiple destinations with proliferation (under $\phi_c = 0, \alpha = 20$): (a) on-time delivery rate, (b) transmission cost incurred for delivery to destination 1, and (c) transmission cost incurred for delivery to destination 2.

Acknowledgements

The authors would like to thank the Wireless and Sensor Networks Lab (WnSN) at Shanghai Jiao Tong University for sharing valuable data on Shanghai taxi GPS traces.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (no. NRF-2018R1A2B6004006).

ORCID iDs

Seona Lee https://orcid.org/0000-0002-8529-3417
HyungJune Lee https://orcid.org/0000-0003-4655-4298

References

1. Bilal SM, Bernardos CJ and Guerrero C. Position-based routing in vehicular networks: a survey. *J Netw Comput Appl* 2013; 36(2): 685–697.
2. Sharef BT, Alsaqour RA and Ismail M. Vehicular communication ad hoc routing protocols: a survey. *J Netw Comput Appl* 2014; 40: 363–396.
3. Peng Y, Abichar Z and Chang JM. Roadside-aided routing (RAR) in vehicular networks. In: *Proceedings of the 2006 IEEE international conference on communications*, vol. 8, Istanbul, 11–15 June 2006, pp.3602–3607. New York: IEEE.
4. Shrestha RK, Moh S, Chung I, et al. Vertex-based multi-hop vehicle-to-infrastructure routing for vehicular ad hoc

networks. In: *Proceedings of the 2010 43rd Hawaii international conference on system sciences*, Honolulu, HI, 5–8 January 2010, pp.1–7. New York: IEEE.

5. Cha SH, Lee KW and Cho HS. Grid-based predictive geographical routing for inter-vehicle communication in urban areas. *Int J Distrib Sens N* 2012; 8(3): 819497.
6. Giudici F and Pagani E. Spatial and traffic-aware routing (STAR) for vehicular systems. In: *Proceedings of the international conference on high performance computing and communications*, Sorrento, 21–23 September 2005, pp.77–86. Berlin: Springer.
7. Jerbi M, Senouci SM, Meraihi R, et al. An improved vehicular ad hoc routing protocol for city environments. In: *Proceedings of the 2007 IEEE international conference on communications (ICC'07)*, Glasgow, 24–28 June 2007, pp.3972–3979. New York: IEEE.
8. Qureshi KN, Abdullah AH and Lloret J. Road perception based geographical routing protocol for vehicular ad hoc networks. *Int J Distrib Sens N* 2016; 12(2): 2617480.
9. Ding Y and Xiao L. SADV: static-node-assisted adaptive data dissemination in vehicular networks. *IEEE T Veh Technol* 2010; 59(5): 2445–2455.
10. Zhao J and Cao G. VADD: vehicle-assisted data delivery in vehicular ad hoc networks. *IEEE T Veh Technol* 2008; 57(3): 1910–1922.
11. Li G, Boukhatem L and Wu J. Adaptive quality-of-service-based routing for vehicular ad hoc networks with ant colony optimization. *IEEE T Veh Technol* 2016; 66(4): 3249–3264.
12. Liu K, Ng JK, Lee V, et al. Cooperative data scheduling in hybrid vehicular ad hoc networks: VANET as a software defined network. *IEEE/ACM T Network* 2016; 24(3): 1759–1773.
13. Qi W, Song Q, Wang X, et al. SDN-enabled social-aware clustering in 5G-VANET systems. *IEEE Access* 2018; 6: 28213–28224.
14. Tang Y, Cheng N, Wu W, et al. Delay-minimization routing for heterogeneous VANETs with machine learning based mobility prediction. *IEEE T Veh Technol* 2019; 68(4): 3967–3979.
15. Karp B and Kung HT. GPSR: greedy perimeter stateless routing for wireless networks. In: *Proceedings of the 6th annual international conference on mobile computing and networking*, Boston, MA, 6–11 August 2000, pp.243–254. New York: ACM.
16. Naumov V, Baumann R and Gross T. An evaluation of inter-vehicle ad hoc networks based on realistic vehicular traces. In: *Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*, Florence, 22–25 May 2006, pp.108–119. New York: ACM.
17. Tian J, Han L and Rothermel K. Spatially aware packet routing for mobile ad hoc inter-vehicle radio networks. In: *Proceedings of the 2003 IEEE international conference on intelligent transportation systems*, vol. 2, Shanghai, China, 12–15 October 2003, pp.1546–1551. New York: IEEE.
18. Zhu W, Gao D, Foh CH, et al. A collision avoidance mechanism for emergency message broadcast in urban VANET. In: *Proceedings of the 2016 IEEE 83rd vehicular technology conference (VTC spring)*, Nanjing, China, 15–18 May 2016, pp.1–5. New York: IEEE.
19. Jiang X and Du DH. PTMAC: a prediction-based TDMA MAC protocol for reducing packet collisions in VANET. *IEEE T Veh Technol* 2016; 65(11): 9209–9223.
20. Lochert C, Mauve M, Füßler H, et al. Geographic routing in city scenarios. *ACM SIGMOBILE Mob Comput Commun Rev* 2005; 9(1): 69–72.
21. Vahdat A and Becker D. Epidemic routing for partially-connected ad hoc networks, 2000, http://issg.cs.duke.edu/epidemic/epidemic.pdf
22. Spyropoulos T, Psounis K and Raghavendra CS. Spray and wait: an efficient routing scheme for intermittently connected mobile networks. In: *Proceedings of the 2005 ACM SIGCOMM workshop on delay-tolerant networking*, Philadelphia, PA, 22 August 2005, pp.252–259. New York: ACM.
23. Wu D, Zhu G and Zhao D. Adaptive carry-store forward scheme in two-hop vehicular delay tolerant networks. *IEEE Commun Lett* 2013; 17(4): 721–724.
24. Lo WZ, Gao JS and Lo SC. Distance-aware routing with copy control in vehicle-based DTNS. In: *Proceedings of the 2012 IEEE 75th vehicular technology conference (VTC spring)*, Yokohama, Japan, 6–9 May 2012, pp.1–5. New York: IEEE.
25. Cheng PC, Lee KC, Gerla M, et al. GeoDTN + NAV: geographic DTN routing with navigator prediction for urban vehicular environments. *Mob Netw Appl* 2010; 15(1): 61–82.
26. Wu B, Shen H and Chen K. Exploiting active subareas for multicopy routing in VDTNs. *IEEE T Veh Technol* 2017; 67(5): 4374–4388.
27. Ahmed S and Kanere SS. SKVR: scalable knowledge-based routing architecture for public transport networks. In: *Proceedings of the 3rd international workshop on vehicular ad hoc networks*, Los Angeles, CA, 29 September 2006, pp.92–93. New York: ACM.
28. Leontiadis I and Mascolo C. GeOpps: geographical opportunistic routing for vehicular networks. In: *Proceedings of the 2007 IEEE international symposium on a world of wireless, mobile and multimedia networks*, Espoo, 18–21 June 2007, pp.1–6. New York: IEEE.
29. Fernandez JA, Borries K, Cheng L, et al. Performance of the 802.11p physical layer in vehicle-to-vehicle environments. *IEEE T Veh Technol* 2011; 61(1): 3–14.
30. Eichler S. Performance evaluation of the IEEE 802.11p WAVE communication standard. In: *Proceedings of the 2007 IEEE 66th vehicular technology conference*, Baltimore, MD, 30 September–3 October 2007, pp.2199–2203. New York: IEEE.
31. Beasley JE and Chu PC. A genetic algorithm for the set covering problem. *Eur J Oper Res* 1996; 94(2): 392–404.
32. Chvatal V. A greedy heuristic for the set-covering problem. *Math Oper Res* 1979; 4(3): 233–235.
33. SUVnet-Trace Data, Shanghai Jiao Tong University. http://wirelesslab.sjtu.edu.cn/taxi_trace_data.html (accessed 24 October 2018).
34. Li YJ. An overview of the DSRC/WAVE technology. In: *Proceedings of the international conference on heterogeneous networking for quality, reliability, security and robustness*, Houston, TX, 17–19 November 2010, pp.544–558. Berlin: Springer.