

An Adaptable Destination-Based Dissemination Algorithm Using a Publish/Subscribe Model in Vehicular Networks

Mildred Madai Caballeros Morales, Rim Haw, Eung Jun Cho, Choong Seon Hong*, and Sungwon Lee

Networking Lab., Department of Computer Engineering, Kyung Hee University, Yongin, Korea

madai84@gmail.com, rhaw@khu.ac.kr, ejcho@networking.khu.ac.kr, cshong@khu.ac.kr, drsungwon@khu.ac.kr

Abstract

Vehicular Ad Hoc Networks (VANETs) are highly dynamic and unstable due to the heterogeneous nature of the communications, intermittent links, high mobility and constant changes in network topology. Currently, some of the most important challenges of VANETs are the scalability problem, congestion, unnecessary duplication of data, low delivery rate, communication delay and temporary fragmentation. Many recent studies have focused on a hybrid mechanism to disseminate information implementing the store and forward technique in sparse vehicular networks, as well as clustering techniques to avoid the scalability problem in dense vehicular networks. However, the selection of intermediate nodes in the store and forward technique, the stability of the clusters and the unnecessary duplication of data remain as central challenges. Therefore, we propose an adaptable destination-based dissemination algorithm (DBDA) using the publish/subscribe model. DBDA considers the destination of the vehicles as an important parameter to form the clusters and select the intermediate nodes, contrary to other proposed solutions. Additionally, DBDA implements a publish/subscribe model. This model provides a context-aware service to select the intermediate nodes according to the importance of the message, destination, current location and speed of the vehicles; as a result, it avoids delay, congestion, unnecessary duplications and low delivery rate.

Category: Ubiquitous computing

Keywords: Smart-vehicle computing; VANETs; Dissemination; Publish; Subscribe; Clustering; Context-aware

I. INTRODUCTION

Nowadays, Vehicular Ad Hoc Networks (VANETs) are widely used by the applications of intelligent transportation system (ITS) to disseminate information related to public safety (e.g., accident warnings), dynamic local information (e.g., maps, location of the cheapest gas stations, available parking areas) and road warnings (e.g., ice, oil, water, low bridges). VANET requires a reliable communication system to support disconnections in sparse networks and to handle the scalability problem in dense

networks. A VANET communication system should be smart and efficient, following the patterns and behavior of the vehicles to disseminate the information where it is needed at the appropriate moment based on the nearby context, interests and geographic location of vehicles. Dissemination of information requires intermediate nodes (i.e., guards, relay nodes, custodians or carriers) to store and forward the information. Many recent studies have focused on the dissemination of information, using the store and forward technique to provide a reliable communication system. These studies have focused on the distri-

Open Access <http://dx.doi.org/10.5626/JCSE.2012.6.3.227>

<http://jcse.kiise.org>

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 31 August 2012, **Accepted** Accepted 3 September 2012

*Corresponding Author

bution of information over all available paths (e.g., [1-6]) or over specific network paths (e.g., [7-11]). Each study has proposed different criteria for selecting intermediate nodes and disseminating information. Flooding [1] is one of the best known methods of information dissemination but it has several problems such as congestion and high rate of unnecessary duplication of data, due to all the vehicles receive all the data. Some alternatives have considered the selection of intermediate nodes to reduce these problems. The efficiency of these alternatives depend of the methods and variables used in the selection of intermediate nodes; certain solutions (e.g., [5, 9-11]) consider the vehicular density, the direction of the vehicles, the connectivity and the delays of the paths as key variables for the selection of the nodes. Additionally, many investigators have considered the message context to mainly disseminate information to the interested vehicles and to classify the information in order to ensure delivery of the messages within the required time (e.g., [2-4]). However, these studies do not consider the destination of the vehicles and the group mobility patterns. Therefore, they select intermediate nodes based on the current location of the vehicles without knowing what path the vehicles will follow. Furthermore, they do not take advantage of the fact that the vehicles sometimes travel in groups to a certain destination when the dissemination of information occurs.

We propose an efficient *Destination-Based Dissemination Algorithm using a Publish/Subscribe Model* (DBDA), which includes the destination of the vehicles and the group mobility patterns as key variables for the selection of intermediate nodes and dissemination of information. The aim of DBDA is to extend the previous studies by addressing, in a clever way, the disconnection problems in sparse networks and the scalability problem in dense networks, to improve the delivery ratio and to reduce the unnecessary duplication of data, congestion and delivery delay. DBDA provides an adaptable dissemination schema in accordance with the current network conditions. This schema considers the classification of the messages based on the importance of each message and the context of the vehicles and messages. This classification is performed using a match function over a publish/subscribe (pub/sub) model. Furthermore, DBDA leverages the group mobility pattern through a stable clustering algorithm. This algorithm considers the behavior of the vehicles, using the destination, current location and speed of the vehicles as parameters to arrange the clusters.

DBDA behavior was evaluated using a real-world scenario with information of 4,000 taxis in Shanghai city; this information provides realistic vehicular traces to make several simulations. The simulation results reveal outstanding performance in terms of delivery rate, overhead and delivery delay. It could be confirmed that the destination and group mobility patterns are valuable elements for the selection of custodians to improve the delivery

rate and decrease the congestion and delivery delay during the dissemination of information.

The rest of the paper is structured as follows. Section II provides an overview of the approaches related to the dissemination algorithms. In Section III, we explain the scenario, the network model and the assumptions of the proposal. In Section IV, the destination-based dissemination algorithm is described. Section V includes performance and evaluation of the proposal. Section VI discusses other factors of the model. Lastly, Section VII contains our conclusions.

II. RELATED WORK

The selection of custodian nodes is widely recognized as an important and distinctive element for the dissemination of information. There are several methods concerning how to select the custodian nodes and disseminate the information.

The direction-aware function driven feedback-augmented store & forward diffusion (DFD-FSFD) [5] method selects the custodian nodes and disseminates the information based on a propagation function that encodes the destination areas of each message as well as the roads it should follow to reach such areas. The function can be determined based on information regarding average road traffic. It is a spatial-temporal relevance function to limit the diffusion of advertisements. Messages should be steered towards the areas within the communication range in which the function returns the lowest values. The route to the destination is the result of the function at each routing hop. DFD-FSFD evaluates the angle between the direction of node movement and the gradient of the propagation function. However, DFD-FSFD is concentrated on the direction of the vehicles rather than the destination of the vehicles.

Context assisted routing (CAR) [8] addresses the disconnections (i.e., permanent topology holes issue) by converting the road infrastructure into a weight-oriented graph. The graph is used in the forwarding decisions and in the predictive location updating algorithm. CAR proposed a mobile network partitioning method in square geographic regions, and nodes in the region are responsible for generating, receiving, storing and disseminating specific traffic context data (static or dynamic data) in the region. Likewise, the connected dominating set (CDS) [9] used a graph with junctions of the roads to explore the possibility of information exchange at the intersections of road segments where different roads join each other. In order to disseminate the information, CDS implements a query message method, taking into account the direction and speed of the vehicles, as well as the density and the connectivity of the path. CAR and CDS utilize geographic forwarding by mainly focusing on network connectivity but they are not designed to target delay

sensitive applications.

Whereas, connectivity-aware minimum-delay geographic routing (CMGR) [10] considers the communication delay, adapting the idea of a beaconing period. CMGR also assumes that vehicles are equipped with digital maps that include detailed locations of streets and junctions (i.e., graph of the road). This method selects the most appropriate route according to the direction of the vehicles, the delay, bandwidth and connectivity of the route, assuming that the density along the path does not change. Another example of a method designed to target delay sensitive applications is PROMPT [11]. This method determines the paths by considering the total path delay and fairness, not just the network connectivity. Additionally, it implements a packet train technique where the transmitter can bundle multiple packets to the same destination and network traffic statistics. PROMPT employs traffic statistics to gather information of the network connectivity but the statistics become stale after a time due to potential changes in route selection and traffic demands.

These methods have emphasized maintaining the path by which information is sent, considering the geographic position, density, connectivity and delay of the paths to select the path. On the other hand, in [2-4] the information is sent through available paths, and the content of the message (e.g., customer preferences and vehicles priorities) is considered when sending the information.

Kosch et al. [2] proposed a method, which takes into account different contexts for a benefit-oriented message dissemination scheme. This method computes the relevance of the messages based on the vehicle context (e.g., the current situation of the vehicular network, current connectivity of the VANET), the message context (e.g., the amount of time elapsed since the message was generated), and the information context (e.g., the interest that other vehicles may have in the information). Similarly, vehicular event sharing peer-to-peer architecture (VESPA) [4] considers the relevance of the messages to drivers based on the concept of encounter probability (EP). EP considers the spatial and the temporal aspects of the information to implement a classification of events. The calculation process of EP includes the distance between the vehicles and the position of the event, the delay, aging penalty and direction. If the EP value obtained is greater than a certain diffusion threshold, the message is rebroadcast. However, these methods usually focus on the direction of the vehicles, rather than the destination and group mobility of the vehicles. The direction is not as accurate as the destination of the vehicles is.

In contrast, hybrid publish-subscribe [3] takes into account the destination of the vehicles, proposing a pub/sub model to disseminate information mainly to subscribers without affecting non-interested vehicles. It computes the utility of the vehicle v with respect to a message m and applies the Dijkstra algorithm to estimate the minimum time (i.e., delay) to reach the home zone. The utility

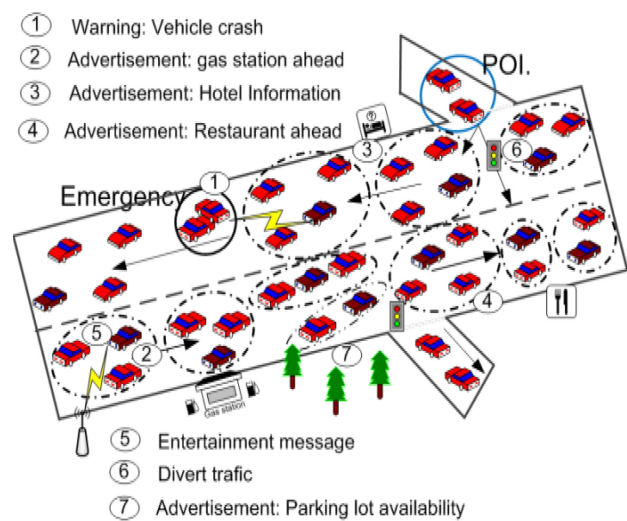


Fig. 1. Urban scenario. POI: point of interest.

describes how good a carrier v is at delivering m to its home zone. Although the destination is considered when calculating the delay, little attention has been paid to the group mobility patterns and classification of the messages.

III. SCENARIO

VANET connects the on board units (OBU) of the vehicles with other OBUs and with the fixed communication devices on the road such as base stations (BS). Fig. 1 illustrates a realistic VANET scenario using an ITS. In this system the vehicles exchange messages according to their interests and their current location. As can be seen in the following cases:

- Message 1 is a warning about a vehicle crash; this warning has to be disseminated to all the vehicles near the collision zone and the point of interest (POI).
- Messages 2, 3, and 4 are advertisements of a gas station, hotel and restaurant, respectively, which should be sent only to interested vehicles.
- The advertisement of parking lot availability (i.e., message 7) is disseminated similar to others advertising messages with the difference that it contains more dynamic information due to the short duration of parking availability.
- Message 5 includes entertainment information, which is sent from the BS to the interested vehicles.
- The traffic light diverts the traffic to avoid congestion in the collision zone when it receives message 6 from the local server.

Therefore, the scenario includes both vehicular to vehicular (V2V) and vehicular to infrastructure (V2I) communications.

The messages are rebroadcast by vehicles to reach the

target location or POI, sending the messages to vehicles that probably are not interested. In dense networks, this fact causes the scalability problem, duplication of data and congestion. On the other hand, in sparse networks, it could cause delivery delay and loss of critical data. In order to address these problems, we propose an effective dissemination algorithm based on a clustering formation. This algorithm considers the importance of each message, the context of the environment and the destination of the vehicles to propagate the messages and select the custodian nodes.

A. Assumptions

- In the network three entities are considered: mobile nodes (MN), cluster head (CH), and BS.
- Each vehicle has onboard units with a global position system (GPS) or navigation system (NS).
- The destination of each vehicle is known. This assumption is realistic for real-world systems because GPS or NS allows drivers to enter their travel destination.
- Historical information on the traffic and density of each street is available.
- The communication is content-based.
- The routing is geographic-based.
- To transmit data efficiently, a combination of broadcast data transmission as well as store-and-forward is used.
- A publish/subscribe strategy is utilized, in which the CH can be both the information provider (publisher) and the information consumer (subscriber).
- The requests to the BS and the dissemination of data to the cluster members are the responsibility of the CH.
- Vehicles periodically (at each Δt) exchange information to choose the CH and the affiliations to each cluster.

B. Road Maps

We address the locations (x, y) on the road through regions. Each region comprises a part of the city formed by streets and avenues. All the points that belong to a region are mapped to a coordinate system (x, y) as shown in Fig. 2, in which a region of Suwon, South Korea, with a dimension of 500 × 500 square meters, is mapped to the coordinate system.

We map the position (longitude, latitude and angle) of the vehicles, provided by the GPS or NS, to points (x, y) on the road within a specific region [13]. When a vehicle enters in a new region, it communicates with the local sever to match its current position and destination with the coordinate system of the region.

The regions and vehicles are identified with IDs. The IDs maintenance is addressed through the counting bloom filter (CBF) [9]. Fig. 2 presents an example of IDs

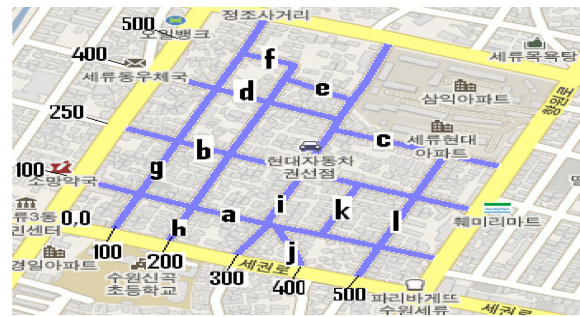


Fig. 2. Region of Suwon, South Korea.

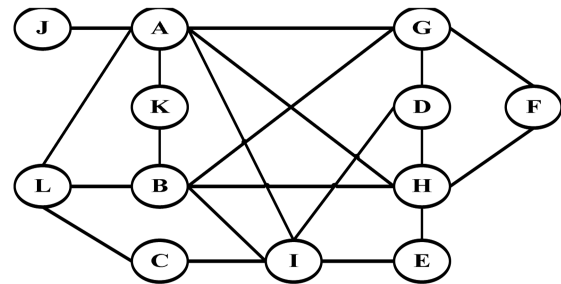


Fig. 3. Graph of junctions.

in which letters from “a” to “l” identify each road.

In some cases specific maps of the roads and junctions of the regions are available. These maps include statistics regarding the density of the junctions. These statistics provide information to draw a detailed graph of the junctions.

Fig. 3 demonstrates the junctions graph for the Suwon map, in which each vertex or circle represents a road and each edge or line represents junctions between two roads $G = \{V, E\}$. This graph is constructed in a similar way to the graphs proposed in CAR [8], CDS [9], and CMGR [10]. However, we incorporate the range of location (x, y) of the road and junctions (i.e., $V = \{id, x, y, density\}$ and $E = \{id, x, y, density\}$) for each region. This range of locations in each vertex and edge is used to compare the destinations of the vehicles, target zones and paths of the messages. The goal of these comparisons is to only send the data for paths that are more efficient in terms of distance and vehicular density. Thus, the main purpose of the junctions graph is the selection of the best paths.

IV. DISSEMINATION ALGORITHM

The proposed dissemination algorithm works with two important systems, the clustering formation and the publish/subscribe strategy. The general architecture for the dissemination algorithm is shown in Fig. 4. The architecture is divided into three main modules (i.e., *Services*, *Management* and *Server Provider* modules).

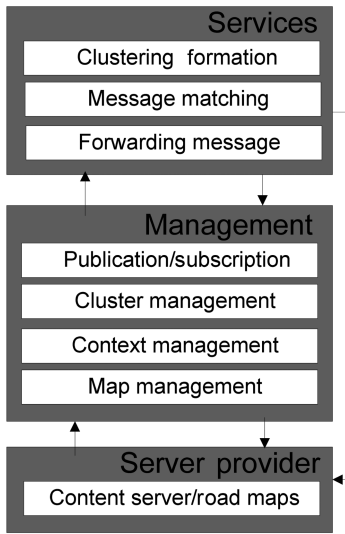


Fig. 4. Architecture.

The *Services* module directly provides the cluster formation, message matching and forwarding message services to the requestors. The requestors are vehicles in general, both CH and MN. The entity who provides the services is usually the CH. However, in some cases these services could be directly provided by the *Service Provider*.

The *Management* module addresses all the mechanisms, which help adequately disseminate the information. These mechanisms are grouped into four sub-modules: publication/subscription, cluster management, context management and map management. The publication/subscription mechanisms support the message matching and forwarding message services. Cluster management is responsible for cluster formation and maintenance.

Context management and map management provide services to the publication/subscription and cluster management. The context management addresses the mobility of the vehicles, statistics and context-aware services such as the location of the vehicles (x, y), services, and events of a specific location. The map management is responsible for managing the information related to the detailed maps.

Finally, the *Server Provider* module provides the content and road maps.

A. Clustering Formation

The clustering algorithm operates with the current location of the vehicles in terms of (x, y), speed, relative destination and final destination (Relative destination: it is the nearest destination according to the current region. Final destination: it is the final destination which is registered in the GPS or NS) in terms of (x, y) as parameters to arrange the clusters and select the CH. These parameters are used to calculate the relative values between two vehicles V and Z ($F_{v,z}$) [14] as shown in Equation (1).

$$F_{v,z} = w_1 \times \Delta L_{v,z} + w_2 \times \Delta S_{v,z} + w_3 \times \Delta RD_{v,z} \quad (1)$$

where w_1, w_2, w_3 represent the weights for each variable, $\Delta L_{v,z}$ is the distance between V and its neighbor node Z , $\Delta S_{v,z}$ is the difference in speed between V and its neighbor node Z and $\Delta RD_{v,z}$ is the distance between the relative destination of V and the relative destination of Z . The value $F_{v,z}$ should be calculated for all vehicles Z within the transmission range (m) of V . These values are used to arrange the clusters. In order to select the CH for each cluster, the F_v value is calculated as shown in Equation (2), where n is the number of vehicles that belong to a specific cluster. The selection of the CH is performed based on the minimum F_v value. Therefore, the vehicle that has the minimum F_v value within a cluster will be nominated as CH [14].

$$F_v = \sum_{i=1}^n F_{v,z} \quad (2)$$

In an effort to address the high mobility of VANETs, cluster maintenance is executed. Cluster maintenance comprises the add member, delete member, update mobility changes, change of CH, and the re-clustering opera-

Algorithm 1 Adding members

PROCEDURE Add Member (MN)

BEGIN

Require:

Mobile node: MN

The MN's list of nearby CHs: L

The selected CH: CH_S

Precondition:

Initialize the L

Process:

If L is empty

The MN nominates itself as the CH;

Else

CH_S = MN selects best candidate of the L;

MN sends an "add" message to CH_S;

CH_S computes the F_v value of MN;

If MN 's F_v value \geq CH_S 's F_v value

//CH_S adds the information of the MN

If CH_S .members $\leq d_{max}$

CH_S.add_member(MN);

CH_S sends a "member_ack" message to MN;

Else

MN selects other CH

Endif

Else

//MN becomes the new CH

CH_S sends the "CH_ack" message to MN;

CH_S sends its Parameter table to MN;

MN.change_CH(CH_S);

CH sends the "member_update" message to all the members of the cluster;

Endif

Endif

END

tions. These operations are presented from Algorithm 1 to Algorithm 5.

The thresholds related to the density, speed and bandwidth of vehicles are considered in the algorithms.

- Density: The cluster should not be big because too many node members may cause overhead, nor too small because very few node members in the cluster increases the number of re-affiliations.

d_{max} : Maximum number of vehicles allowed in a cluster.

d_{min} : Minimum number of vehicles allowed in a cluster.

CH_{max} : Maximum amount of CHs that will form the neighboring cluster in the re-clustering operation.

- Speed:

ΔS_{max} : Maximum limit for speed variance.

- Bandwidth:

bw_{min} : Minimum bandwidth or QoS level required.

Add member

- Step 1: *Find the candidate CH.* The MN selects the best candidate from its own list of nearby CHs. The selection of the CH depends on which has the minimum value of F_{vz} between the CH candidates and the MN.
- Step 2: *Add request.* The MN sends an “add” message to the selected CH.
- Step 3: *Computation of the F_v value.* When the CH receives the “add” message, it computes the F_v value of the MN and compares it with its own value.
- Step 4: *Add the MN.* Depending on the F_v values, two cases might occur.

Case 1: The F_v value of the MN is not less than the F_v value of the CH. Therefore, the CH adds the information of the MN to the cluster. The CH

Algorithm 2 Deleting members

PROCEDURE Delete Member (MN)

BEGIN

Require:

Mobile node: MN

Current CH: CH

Parameter table of the current CH: P

Process:

If the MN is within CH’s transmission range

MN sends a “delete” message to its CH;

CH.delete_member(MN);

//CH erases the MN’s information from P

If CH.members < d_{min}

Reclustering(CH);

Endif

Else

If the communication between CH and MN fails

CH.delete_member(MN);

//MN looks for another CH;

Add Member(MN);

Endif

Endif

END

checks the d_{max} value and the current quantity of members to avoid violation of the threshold. The CH adds the MN and replies with a “member_ack” message. If a violation of the threshold exists, the MN has to select another candidate to be the CH, starting from step 1.

Case 2: The F_v value of the MN is less than the F_v value of the CH. Thus, the MN becomes the new CH. The CH sends the updated information of all the members with the “CH_ack” message to the MN; consequently, the MN is the new CH. The new CH sends the “member_update” message to all the cluster members.

If the MN cannot locate a CH, it will nominate itself as the CH, forming a new cluster.

Delete member

A node member is removed from the cluster if any of the followed cases happen:

- Case 1: The MN is within the transmission range of the CH.

Step 1: *Delete request.* The MN sends a “delete” message to conclude its subscription with the current CH.

Step 2: *Erase MN.* The CH erases the information of the MN and compares the current number of

Algorithm 3 Updating the mobility changes

PROCEDURE Update Mobility (E)

BEGIN

Require:

Entity is a MN or CH: E

CH of the entity E: CH

Changes in Speed: S

Changes in Bandwidth: B

Number of warning messages: W

Precondition:

Initialize W

Process:

For each Δt **do**

If E.S > ΔS_{max} or E.B < bw_{min}

E sends a “speed_warning” and/or

“bandwidth_warning” to its CH;

If E is a MN

Add Member(E.MN);

Delete Member(E.MN);

Endif

W=CH.warnings();

If W is significantly high

ChangeCH(CH.);

Endif

Else

E recalculates the average speed;

E saves the average speed and bandwidth into its historical information;

Endif

Endfor

END

members against the d_{min} value.

- Case 2: The MN is out of the transmission range of the CH.

When a MN is unreachable, both attempt to communicate during the time to live. If the communication fails, the CH deletes the MN's information and the MN starts to look for another CH.

Update process of the mobility changes

For every Δt , each entity (CH or MN) verifies if there are changes in the average speed and the bandwidth. If the absolute change in speed is less than or equal to ΔS_{max} the entity recalculates the average speed value and it saves the new value. If the change in speed is greater than ΔS_{max} or the latest bandwidth measure is less than bw_{min} the following tasks are executed, by entity:

- MN sends a "speed_warning" or "bandwidth_warning" message to the CH. The MN compares the $F_{v,z}$ value of the current CH with the $F_{v,z}$ values of the list of nearby CH. If the value is less than the current $F_{v,z}$ value, the MN is affiliated with the new CH.
- CH registers the number of "speed_warning" and "bandwidth_warning" messages received at Δt interval. CH discards repeated messages. If the number of warning messages is significantly high, CH performs the change of the CH.

Change of CH

- Step 1: *Computation of the F_v values.* Calculate the F_v

Algorithm 4 Change of CH

PROCEDURE ChangeCH (CH)

BEGIN

Require:

Current CH: CH

Parameter table of the current CH: P

MN with minimum F_v value: MN

Precondition:

Update P

Process:

For each MN of P

CH.compute $F_v(P.MN)$;

Updates minimum F_v value;

Endfor

MN=P.get_MN_with_minimum_ F_v ;

If minimum $F_v < CH_F_v$

//Select new CH

CH sends the "CH_ack" message to MN;

CH sends its Parameter table to MN;

//MN becomes the CH

MN.change_CH(CH);

CH sends the "member_update" message to all the members of the cluster;

Endif

If (P.size > d_{max} or P.size < d_{min})

Reclustering(CH);

Endif

END

Algorithm 5 Re-clustering

PROCEDURE Reclustering (CH)

BEGIN

Require:

Current CH: CH

CH's list of nearby CHs: L

The list of neighboring clusters: N

Precondition:

Initialize the L

Process:

If L is empty

Cluster Formation(CH);

Else

N=CH.select_neighboring(L);

For each CH of N

//N.size $\leq CH_{max}$

CH sends a "reclustering" message;

N.CH sends a "reclustering_ack" and its

Parameter table to CH;

CH updates its Parameter table merging the

N.CH information;

Calculate $F_v(CH)$;

FindingCH(CH);

Endfor

Endif

END

values for all the MN members of the cluster.

- Step 2: *Select new CH.* If the current CH is not the one with the minimum F_v value, it sends the updated information of the cluster members and the "CH_ack" message to the MN with the minimum F_v value. This MN becomes the new CH and it sends a "member_update" message to all the MN in the cluster.
- Step 3: *Threshold verification.* A verification procedure for the number of members within the cluster is performed, checking the limits d_{min} and d_{max} .

Re-clustering process

The re-clustering process performs the merging and the division of clusters. Throughout the clustering process, there are some events that determine when to make the re-clustering process.

- Event 1: *Deleting a cluster member.* After the CH deletes a member, it checks the number of members, if the amount of members is less than the d_{min} value the re-cluster is mandatory.
- Event 2: *Change of CH.* If the amount of the member is less than the d_{min} value or greater than the d_{max} value the re-clustering process is executed.

Re-clustering process includes the follow steps:

- Step 1: *Select the neighboring clusters.* The CH which initiates the process selects from its list of nearby CHs the neighboring clusters that will participate in the process. The maximum quantity of CHs must not exceed the CH_{max} value.
- Step 2: *Re-clustering request.* The selected CH sends

his members information and a “reclustering_ack” message, only to the CH which initiated the process. All the information is merged by the CH, which initiated the process.

- Step 4: *Computation of the F_v values.* The collected information is used to calculate the F_v values.

If the CH which initiated the process does not have any near CH, the re-clustering process turns into a cluster formation process.

B. Publish/Subscribe Strategy

The subscriptions are divided into automatic and custom subscriptions [3, 4]. In [4], the automatic subscriptions are the events relevant to the vehicles, and custom subscriptions are the events relevant for the drivers. Custom subscriptions are specific interests of the drivers, such as, information regarding nearby restaurants, gas stations, parking or hotels. Automatic subscriptions are calculated depending on the current location. For example, the vehicle will subscribe to an accident or traffic warning service that affects the current route.

The CH has a list of the other nearby CH’s, a list of its available publications and a list of the subscriptions (automatic and custom).

1) Publication/Subscription Maintenance

The publications and subscriptions are constantly changing based on the current context or environment. Therefore, it is possible to add and delete subscriptions and publications dynamically.

- Add subscription: When the CH receives a new subscription from some of its members, it adds the new subscription to the list.
- Delete subscription: The CH deletes the subscription from its subscription list if the subscription is not required for any member.
- Add publication: When an event (e.g., accident) comes up, the witness node sends a message to its current CH to publish the event.
- Delete publication: At each Δt , the CH checks the time to live (TTL) of the publications to eliminate obsolete publications from its publication list and verifies the diffusion area, which is based on the home location, the target location of the publication, and the current location of the CH. The TTL and diffusion area are used to eliminate publications that are not of interest.

In order to generate useful data when two CHs start the message exchange, both CHs share their current publication messages.

2) Match Function

In order to determine which messages match with the interests of the subscriptions, the CH implements a match function.

The execution of the match function constitute the following steps:

- *Aging threshold:* Compares the aging threshold of the message against the time that the message has traveled in the network. Δa quantifies this comparison through Equation (3) where “t” is the current time. The equation shows two cases. When the time that the message is traveling in the network (i.e., current time (t) minus creation time of the message (CT), (t – CT)) is less than or equal to the TTL, the value of Δa is between 0 and 1; otherwise the value of Δa is greater than the value of TTL [15].

$$\Delta a \begin{cases} \frac{t-CT}{TTL}; (t-CT) \leq TTL, \\ 1-CT, \text{otherwise.} \end{cases} \quad (3)$$

Equation (3) ensures the penalty for all messages that surpass the TTL value.

- *Priority:* It is the calculation for the priority of the message which is a value between 1 to n.
- *Distance threshold (Δd):* The calculation of Δd requires two values: the distance from the target location of the message to the current location and to the relative destination of the vehicle.
 - Distance to the current location ΔCL : It is the difference in distance between the current location of the vehicle (CL) and the target location (TL) in terms of (x, y) coordinates [15], calculated using Equation (4).

$$\Delta CL = \sqrt{(CL_x - TL_x)^2 + (CL_y - TL_y)^2} \quad (4)$$

- Distance to the relative destination ΔD : It is the difference in distance between the relative destination of the vehicle (RD) and TL in terms of (x, y) coordinates [15]. This difference can be calculated with the information from maps as shown in Equation (5).

$$\Delta D = \sqrt{(P_x - TL_x)^2 + (P_y - TL_y)^2}, P \in path \quad (5)$$

where P is the nearest point to the TL. This point is part of the path of the vehicle. However, in some remote areas, the availability of maps is limited. In this case ΔD is calculated by employing the relative destination [15].

$$\Delta D = \sqrt{(RD_x - TL_x)^2 + (RD_y - TL_y)^2} \quad (6)$$

This calculation of Δd emphasizes the relevance of the messages when the vehicles are within the radius R of the target zone for the message [15] as shown in Equation (7).

$$\Delta d \begin{cases} \frac{\Delta CL}{R}; \Delta CL \leq R, \\ \Delta D, \text{otherwise.} \end{cases} \quad (7)$$

The value of Δd is calculated for each TL by selecting the minimum Δd .

- **Match Subscription (MS):** The match subscription is a binary value, 0 when no subscription matches the message and 1 when at least one subscription matches the message. If the message belongs to the custom category, the match subscription is 1 if $\Delta CL \leq R$; otherwise it is 0. On the other hand, if the message belongs to the automatic category, the match subscription is 1 if at least one of the lists of subscriptions has the same type of message.

Finally, the relevance of the messages is calculated by taking into account the ageing threshold, the priority, the distance threshold and the match subscription [15].

$$Relevance = w_1\Delta a + w_2Priority + w_3\Delta d + w_4MS \quad (8)$$

In Equation (8) the weights w_1 , w_2 , w_3 , and w_4 determine the importance of each variable. The messages with the lowest “Relevance” values are the most important messages. The CH selects the messages to be published or deleted according to the “relevance” value. This selection process aims to deliver only the messages that are important or interesting to each vehicle, at the most opportune time in order to avoid congestion, delay and duplication of data.

C. Dissemination Algorithm

The basic steps of the dissemination algorithm are shown in Fig. 5. The step to match the message against the subscription is implemented by the match function providing the relevance value of the message. This relevance is used to rebroadcast the message to the interested vehicles [12].

1) Message Matching

After the CH obtains the relevance value, the CH completes the verification to decide if the message will be published (forwarded to its cluster members), forwarded to its neighbor nodes or dropped, by checking the following thresholds:

- **Relevance Threshold (RT):** This threshold determines which messages are important for the cluster members. If the relevance value of the message is less than the threshold, the message has to be published and

forwarded to the cluster members.

- **Storage Threshold (ST):** Before publishing a message, the CH verifies if the available storage is greater than or equal to the storage threshold. When the available storage is less than the storage threshold, the CH deletes the publication with the highest relevance value and adds the new message. Furthermore, the deleted publication is forwarded to all the nearby CHs, if the condition presented in Equation (9) is true.

$$(t - CT) > TTL \quad (9)$$

- **Dissemination Threshold (DT):** The goal of this threshold is to dynamically adjust the algorithm according to the vehicular density in order to avoid congestion in dense networks and packet loss in sparse networks. The CH determines the number of

Algorithm 6 Message matching

PROCEDURE Message Matching (m)

BEGIN

Require:

Message: m

Relevance value of the message m : Rm

Current CH: CH

Publication List of the current CH: PL

Available Storage of the CH: $CH.AS$

Message of the PL with the highest Relevance value:

m_Rmax .

Process:

$Rm = CH.match_function(m)$;

If $Rm \leq RT$

If $CH.AS \geq ST$

$PL.add(m)$;

Else

$m_Rmax = PL.get_highest_relevance()$;

$PL.delete(m_Rmax)$;

$PL.add(m)$;

If $(m_Rmax.t - m_Rmax.CT) > m_Rmax.TTL$

Forward the m_Rmax to all the nearby CHs

Endif

Endif

Else

If $((m.t - m.CT) > m.TTL \text{ and } (m.Number_retransmissions \leq DT))$

Forward the m to all the nearby CHs

Endif

Endif

END

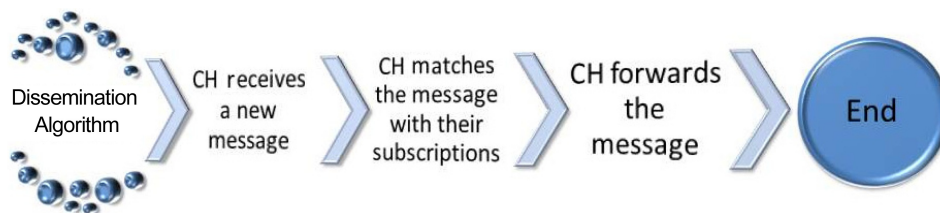


Fig. 5. Dissemination algorithm. CH: cluster head.

retransmissions in regards to the messages depending on the vehicular density.

Algorithm 6 illustrates how the CH executes the process of message matching.

2) Forwarding Message

In an effort to rebroadcast or propagate messages efficiently, we consider two scenarios. 1) Information is propagated within the TL or POI, and 2) information is propagated out of the TL (delivery-IN and delivery-OUT [5]). In the first scenario, the goals are to send the automatic subscriptions to all vehicles (i.e., broadcast) within the CL and send the custom subscriptions only to the interested vehicles using the "Relevance" value. A variation of the flooding dissemination is implemented, in which the CH sends the messages to all its neighbors and each CH rebroadcasts the messages to its clusters members.

The second scenario is more complicated. The CH must select the best carriers or custodians of the messages to reach the TL (POI) (e.g., multicast, geocast, and unicast) and to minimize delivery of messages that are not of interest. Selection for the best carriers includes the following tasks:

a) *Density measures*: The dissemination algorithm takes advantage of road statistics to calculate the density. Vehicular density is an essential factor to ensure a high delivery ratio because we need to select a path with good connectivity that is dense enough to reach the target location but not so dense that it will generate overhead and cause a low bandwidth. Therefore, we used a similar version of the method proposed in CMGR [10] to calculate the connectivity in the network through the average expected value of the density over a certain period of time T ($T\rho_V$). In order to calculate this value, we define the variables that describe the rate of change in the neighboring area of the vehicle V (r_V) and the density in the neighboring area of the vehicle V (ρ_V). r_V is calculated using statistics and ρ_V is calculated using message exchanges as can be seen in Equation (10).

$$\rho_V = \frac{Nvr_V}{Lca_V} \quad (10)$$

where Nvr_V is the number of vehicles in the transmission range of V , and Lca_V is the length of the coverage area.

The density of the neighboring area for vehicle V during a certain period of time T ($T\rho_V$) is calculated through Equation (11).

$$T\rho_V = \rho_V + r_V(T) \quad (11)$$

Additionally, the number of retransmissions for messages can be adjusted according to the density and the priority of the messages through a for-

warding method, which only rebroadcasts updates of information at the junctions (graph-based predictive location service) [8] when dense scenarios and low priority messages are present. Furthermore, when two vehicles are rebroadcasting the same messages, the forwarding method selects the vehicle which has a better location (x, y) with respect to the TL to rebroadcast the message. When this occurs, the vehicle with the worst location stops re-diffusion of the message.

b) *Path management*: In an effort to avoid congestions caused by flooding dissemination, vehicles can forward the data only through specific paths on the road. Selection of the paths is implemented with information provided by the road maps (e.g., number of lanes, junctions in the road). This method is similar to the method presented in CDS [9] and CAR [8]. However, we consider the (x, y) location of each junction in order to select the best path in terms of the shortest distance to the POI and average speed (The Dijkstra algorithm is used to calculate the shortest path and highest connectivity). To improve the performance of the algorithm, it is possible to implement an aggregation algorithm such as TrafficView [16] to disseminate the information in bundles.

c) *Custody transfer*: For the purpose of reducing the packet loss and delivery delay in sparse networks when there is no path having sufficient connectivity, the algorithm implements a store and forward technique by selecting a custodian to carry the messages. The selection of the custodian depends on the relevance of the message as well as the current speed and storage capacity of the vehicle. In order to reduce the delivery delay of important messages, critical messages have priority over non-critical messages. When a custodian of a specific message is leaving the POI; it has to transfer the message to another custodian. In an effort to ensure the delivery of the messages, an acknowledge mechanism is performed. In this mechanism the current custodian has to carry the message until it finds another custodian which assumes responsibility for the message. In addition, we implement a carry-and-forward network (CafNet) [17] to take advantage of the infrastructure. Thus, messages are sent across an intermittently connected network through the BS or hot spots similar to the VPN proposed in CAR [8]. However, in dense networks, our solution avoids the formation of bottlenecks that can occur in the BS or hot spots by classifying the messages through the Relevance value. This classification prioritizes critical data.

Algorithm 7 summarizes the forwarding method. As can be seen in this algorithm, the message m could be a

Algorithm 7 Forwarding message

PROCEDURE Forwarding Messages (m)

BEGIN

Require:

- Message or bundle of messages: m
- Relevance value of the message m : R_m
- Current CH: CH
- CH's list of nearby CHs: L
- Density of neighboring area of the CH: $T\rho_V$

Process:

```

CH.get_context();
//If the vehicle is in the POI or target location
If  $\Delta CL \leq R$ 
    If  $m.PT.category = automatic$ ;
        CH.send_to_all_neighbors(m);
    Else
        CH.send_to_subscribers(m);
    Endif
If CH is leaving the POI or target location
    CH.search_custodian(L,m);
Endif
//If the vehicle is not in the POI or target location
Else
If CH is in a junction
    CH.select_path( $T\rho_V, road\_speed, m$ );
End if
    CH.send_best_carriers(L,m);
Endif
END
    
```

message or a bundle of messages with the same target location.

V. PERFORMANCE EVALUATION

We evaluated the proposal through a set of simulations. These simulations were developed in Java JDeveloper since a generic simulator is difficult to use, due to the diversity and flexibility of the scenario.

Fig. 6 illustrates the basic structure of our simulator. The initial input for each module is a group of files, which are loaded when the simulation begins.

The *Road* module includes information related to the road structure, maps and graphs. The road structure was abstracted as x and y coordinates for each region. Basically, the road structure of a region in Shanghai was used for the simulations. Specific information on the avenues and streets of the region was detailed in the file *ROAD*.

Additionally, this module includes detailed maps of the road for the regions. These maps form the junction graph $G = \{V, E\}$.

The *Mobility* module receives dynamic information on the *Road* module such as density statistics, the current coordinate system of the region, and paths to a specific target location through the junction graphs.

The mobility module addresses all instances of vehicles

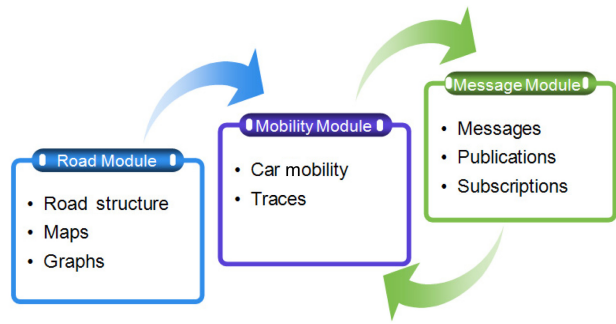


Fig. 6. Simulator structure.

and their movements or traces. The traces can be generated automatically by selecting random initial positions and destinations or generated through input files. The input files include information on the traces of each vehicle.

In order to perform these simulations, we included the information of 4,000 taxis in Shanghai obtained from the Wireless and Sensor Networks Laboratory of Shanghai Jiao Tong University [9]. This data consists of a collection of GPS information that includes the real-time locations of taxis in the urban city of Shanghai. The GPS information includes the ID of the taxi, the longitude, latitude and angle of the current location of the taxi, speed of the taxi, and the time that the GPS record was sent. This information provides real-time information on the taxis, fitting each GPS register to road maps, with interpolation help in cases when the data obtained by GPS did not provide a clear trajectory. Additionally, we modified these data by adding a destination to the taxi's trajectory.

The *Message* module receives the information from the *Mobility* module in order to exchange the messages, arrange clusters and forward messages. The messages, publications and subscriptions can be generated automatically or by input files. The automatic subscriptions are generated according to the current context. Therefore, the message module and the mobility module are constantly interacting.

The general parameters that we used during the simulation are listed in Table 1.

Table 1. Simulation parameters

Parameter	Value
Simulation area	5,000 × 5,000 m
Simulation time	3,600 sec
Average speed	9.75 m/sec
Range speed	0-30 m/sec
Vehicular density	4,000 taxis
Transmission range	300 m
Packet size	1 kb

Each taxi followed its trajectory. Upon reaching its destination, the taxi was removed from the network and inserted from its initial location. This process was repeated during the simulation time. The taxis were grouped in clusters assuming the availability of a detailed road map. The metrics considered in our performance evaluation are: delivery ratio (% of delivered messages), overhead (average number of retransmissions per message), and delivery delay (delay of the delivered message).

A. Simulation Results

In an effort to evaluate the dissemination algorithm, we simulated sparse networks and dense networks, in which a new message is generated every 120 seconds. Forty percent of the messages were custom subscriptions, 40% were non-critical automatic subscription, and 20% were critical automatic subscriptions. The TTL of each message is 300 seconds.

In addition, we included intermittent links [18] to test the behavior of the algorithm in sparse networks. Fig. 7 shows the patterns of the intermittent link over the course of one hour.

For the purpose of this study we assumed a vehicular density between 30 and 35 vehicles per km² during rush hours and between 10 and 15 vehicles per km² during low density hours. This assumption is validated according to the information from the taxis.

Fig. 8 presents the delivery rate percentage of our DBDA compared with the flooding, CDS, and hybrid publish/subscribe and CMGR algorithms. We vary the number of vehicles per square kilometer (km²). This figure exhibits a clear trend; when the vehicular density is low, the delivery rate decreases. The same behavior occurs when the vehicular density is high. This behavior is due to intermittent links at low vehicular density and congestion at high vehicular density. As revealed in Fig. 8, flooding dissemination demonstrated the worst performance because it does not address the disconnections or the scalability problem. On the other hand, with low vehicular density, the other algorithms implement a store and forward strategy. The hybrid publish/subscribe employs a metric based on the delay (time to reach the POI) that is

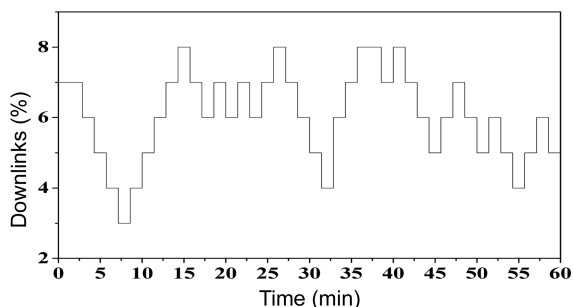


Fig. 7. Number of down links in one hour.

similar to the connectivity metric of the CDS and CMGR. A significant difference is that CDS and CMGR consider the density and delay to have a higher delivery rate than the hybrid publish/subscribe. DBDA employs a context metric based on the relevance of the messages, destination and speed of the vehicles. DBDA also considers the connectivity of the path and takes advantage of the group mobility through the cluster dissemination. Therefore, DBDA had outstanding performance with low vehicular density. When the vehicular density is high, most algorithms had good performance, probably because they avoid congestion. In contrast, the flooding algorithm does not have any control to prevent the scalability problem.

The overhead of the algorithms, in terms of the average number of retransmissions per message, is shown in Fig. 9. As can be seen in this figure, all the methods present an upward trend. This suggests that the overhead is directly proportional to the vehicular density. The DBDA had the lowest overhead for different vehicular densities (i.e., number of vehicles per km²). It is highly likely that the method of forwarding through clusters reduces the overhead. Furthermore, the publish/subscribe strategy (based on the interests of the vehicles) and the method with dynamic adjustment of the retransmission rate based on the vehicular density and the priority of the message also contribute to reduce the overhead in dense networks (e.g., 35 vehicles per km²). Hybrid publish/subscribe, CDS and CMGR had a similar trend in the overhead. However, the slight reduction in overhead regarding hybrid publish/subscribe may be caused by the distinction between the interested and non-interested vehicles.

Flooding dissemination had high overhead. Fig. 9 shows that the overhead increases sharply above a vehicular density of 20 vehicles per km². This is because the vehicles rebroadcast the message to all the neighbors.

Fig. 10 displays the average delivery delay for each message according to the vehicular density of the net-

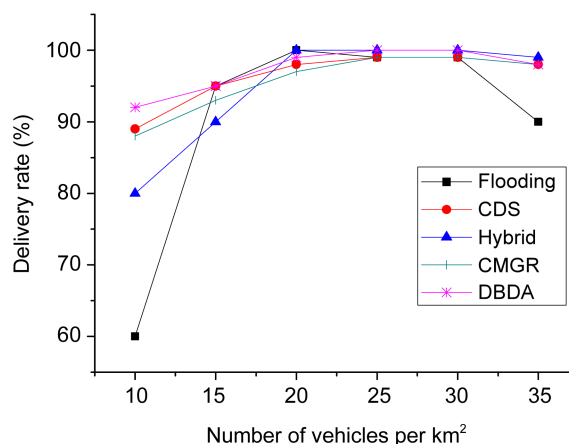


Fig. 8. Delivery rate. CDS: connected dominating set, CMGR: connectivity-aware minimum-delay geographic routing, DBDA: destination-based dissemination algorithm.

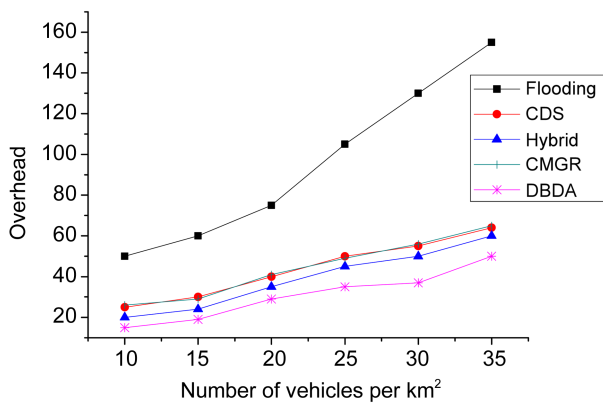


Fig. 9. Overhead (average number of retransmissions per message). CDS: connected dominating set, CMGR: connectivity-aware minimum-delay geographic routing, DBDA: destination-based dissemination algorithm.

work for flooding, CDS, CMGR, DBDA and the DBDA critical method. One of the challenges of these methods is to maintain the delivery delay of the messages as low as possible. However, an overall increase in the delivery delay is inevitable since low vehicular density reduces the opportunity for message exchanges. When the number of vehicles per km² is low (i.e., between 10 and 15 vehicles per km²) the delivery delay is highest. In contrast, when the number of vehicles per km² is between 25 and 35, the delivery delay declined and remained steady.

As revealed in Fig. 10, the flooding method revealed a drastic change in the trend. It had the worst delivery delay (i.e., global maximum) at 10 km² but also had the best delivery delay (i.e., global minimum) at 30 km². This steep fall probably reflects a lack of store and forward technique.

In addition, the delivery delays of CDS and DBDA had similar values, performing better when the vehicular density is high. However, the fall was less steep than with the flooding method. On the other hand, CMGR had a signif-

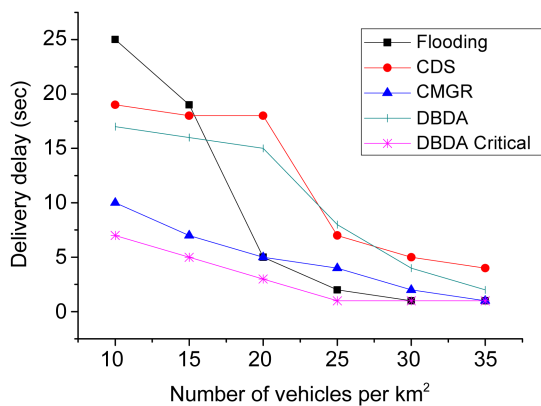


Fig 10. Delivery delay. CDS: connected dominating set, CMGR: connectivity-aware minimum-delay geographic routing, DBDA: destination-based dissemination algorithm.

icant low delivery delay when the vehicular density was low due to infrastructures (gateways). Similarly, DBDA had the lowest delivery delay when it disseminates critical messages (i.e., DBDA critical) and the vehicular density is probably low because it takes advantage of techniques such as CafNet and the match function, which prioritize the dissemination of critical messages.

VI. DISCUSSION

The main purpose of this study is to disseminate information in an appropriate context. This means that the data forwarding method should be able to adapt the message exchanges based on the network conditions in order to distribute the information among the neighbors. For example, data forwarding methods should be based on the importance of the message, density of the road and speed of the vehicles. An appropriate data forwarding method will decrease congestion in dense networks and reduce the low delivery rate in sparse networks.

Overall, the destination-based algorithm (DBDA) provides accurate knowledge in regards to the future situation of the vehicle since the destination determines what information is of interest to the vehicle. Additionally, this algorithm leverages group mobility patterns by considering the destination of the vehicles. The assumption of a known destination is unpopular but nowadays, with advances in technology and the increased use of NS or GPS, drivers can register their destinations. The problem with this assumption is related to encouragement in regards to the willingness of users to input their destinations. In an effort to encourage users to input destination data we suggest an advertising campaign. Furthermore, some proposals include this assumption (e.g., hybrid publish/subscribe [3]). This suggests that the known destination assumption will become popular.

In general, DBDA consists of two aspects, the clustering formation and the publish/subscribe strategy. The clustering formation leverages group mobility to improve resource consumption, load balancing and scalability problems during the exchange of messages. Notwithstanding how high mobility and rapid changes in network topology limit the performance of the clustering algorithm; this study suggests that including the destination of the vehicles will decrease the effects of high mobility and network topology changes. The simulation results reveal that the destination of the vehicles is a key factor in group mobility. Messages are delivered in a clever and efficient way. In addition, the proposed cluster formation could provide more storage capacity to store messages by exploiting the resources for all the members of the cluster. This is important because in reality, storage capacity is limited (cf. contrary to the assumption of unlimited storage capacity presented in hybrid publish/subscribe [3]).

Despite all the advantages provided by clustering algo-

gorithms, some approaches such as VESPA [4] argue that it is difficult for nodes to collect up-to-date neighborhood information. This suggests that the clustering approach might be less effective when neighboring nodes do not have a destination in common or if the speed difference among neighboring nodes is very different. However, in this scenario the performance of DBDA is the same as the performance of proposals which do not consider cluster formation.

On the other hand, the publish/subscribe strategy offers the prioritization of messages considering the location context of the vehicles and the content of the message. According to the simulation results, this strategy decreases the unnecessary duplication of data, congestion and high delivery delay. Furthermore, in an effort to take advantage of the available infrastructure (i.e., BS, hot spots) the information could be sent over a wired infrastructure network, which is similar to the VPN tunnel proposed in CAR [8]. However, in dense scenarios the BS or hot spots can become bottlenecks. The publish/subscribe strategy has addressed bottleneck problems by categorizing the messages according to the relevance of each message. The relevance value of the messages determines which messages are sent through the wired infrastructure. With that in mind, critical messages (i.e., the messages with the lowest-relevance value) have priority.

It should be noted that this strategy has been primarily concerned with categorizing the messages. It would seem, therefore, that further investigations are needed in order to ensure the fairness of the relevance value regarding the messages.

The simulations of this study do not include multiple regions. Therefore, future works could include simulations over multiple regions in order to determine the best size for each region. The size of each region is important because in the selection of the paths, the Dijkstra algorithm is used, and the complexity of the Dijkstra algorithm is $O(n^2)$, where $n = |V|$ is the total number of vertices in the graph. Thus, it is useful to have an optimum value of n to avoid high complexity in the solution of the Dijkstra algorithm.

VII. CONCLUSION

Data dissemination in VANETs requires dynamic and intelligent decisions based on context to work in sparse and dense scenarios in order to ensure reliable communication despite rapid node mobility. We propose a DBDA to address scalability, unnecessary duplication of data and congestion problems in a dense network. DBDA also considers low delivery rate and high delivery delay in sparse networks through the dynamic selection of custodians, cluster-based forwarding and prioritization of messages based on context. The performance evaluation has demonstrated that DBDA offers outstanding performance

by leveraging the destination of vehicles and using road maps to disseminate information. Even in the worst case when there are no available maps, the match function provides good performance by taking message context and mobility into account.

ACKNOWLEDGMENTS

The simulation data were obtained from the Wireless and Sensor Networks Lab. (WnSN), Shanghai Jiao Tong University.

REFERENCES

1. P. Costa, D. Gavidia, B. Koldehofe, H. Miranda, M. Musolesi, and O. Riva, "When cars start gossiping," *Proceedings of the 6th Workshop on Middleware for Network Eccentric and Mobile Applications*, Glasgow, UK, 2008, pp. 1-4.
2. T. Kosch, C. J. Adler, S. Eichler, C. Schroth, and M. Strassberger, "The scalability problem of vehicular ad hoc networks and how to solve it," *IEEE Wireless Communications*, vol. 13, no. 5, pp. 22-28, 2006.
3. I. Leontiadis, P. Costa, and C. Mascolo, "A hybrid approach for content-based publish/subscribe in vehicular networks," *Pervasive and Mobile Computing*, vol. 5, no. 6, pp. 697-713, 2009.
4. T. Delot, N. Cenerario, and S. Ilarri, "Vehicular event sharing with a mobile peer-to-peer architecture," *Transportation Research Part C Emerging Technologies*, vol. 18, no. 4, pp. 584-598, 2010.
5. P. Costa, D. Frey, M. Migliavacca, and L. Mottola, "Towards lightweight information dissemination in inter-vehicular networks," *Proceedings of the 3rd International Workshop on Vehicular Ad Hoc Networks*, Los Angeles, CA, 2006, pp. 20-29.
6. F. Ye, S. Roy, and H. Wang, "Efficient data dissemination in vehicular ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 4, pp. 769-779, 2012.
7. K. Mershad and H. Artail, "Performance analysis of routing in VANETs using the RSU network," *Proceedings of IEEE 7th International Conference on Wireless and Mobile Computing, Networking and Communications*, Wuhan, China, 2011, pp. 89-96.
8. V. Dumitrescu and J. Guo, "Context assisted routing protocols for inter-vehicle wireless communication," *Proceedings of IEEE Intelligent Vehicles Symposium*, Las Vegas, NV, 2005, pp. 594-600.
9. K. Liu, M. Li, Y. Liu, M. Li, X. Y. Li, and L. Guibas, "Chasing the hidden connectivity in urban vehicular networks," 2009.
10. K. Shafiee and V. C. M. Leung, "Connectivity-aware minimum-delay geographic routing with vehicle tracking in VANETs," *Ad Hoc Networks*, vol. 9, no. 2, pp. 131-141, 2011.
11. B. Jarupan and E. Ekici, "PROMPT: a cross-layer position-based communication protocol for delay-aware vehicular

- access networks,” *Ad Hoc Networks*, vol. 8, no. 5, pp. 489-505, 2010.
12. M. M. C. Morales and C. S. Hong, “Destination-based delivery algorithm for dynamic custodian management in VDTN routing,” *Journal of KISS Computing Practices*, vol. 18, no. 2, pp. 110-115, 2012.
 13. X. Li, M. Li, W. Shu, and M. Wu, “A practical map-matching algorithm for GPS-based vehicular networks in Shanghai urban area,” *Proceedings of IET Conferences on Wireless, Mobile and Sensor Networks*, Shanghai, China, 2007, pp. 454-457.
 14. M. M. C. Morales, C. S. Hong, and Y. C. Bang, “An adaptable mobility-aware clustering algorithm in vehicular networks,” *Proceedings of the 13th Asia-Pacific Network Operations and Management Symposium*, Taipei, Taiwan, 2011, pp. 1-6.
 15. M. M. C. Morales, R. Haw, J. Lee, and C. S. Hong, “An efficient destination-based data management policy for vehicular networks,” *Proceedings of the 11th International Conference on ITS Telecommunications*, St. Petersburg, Russia, 2011, pp. 399-404.
 16. T. Nadeem, S. Dashtinezhad, C. Liao, and L. Ifode, “TrafficView: traffic data dissemination using car-to-car communication,” *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 8, no. 3, pp. 6-19, 2004.
 17. U. Lee and M. Gerla, “A survey of urban vehicular sensing platforms,” *Computer Networks*, vol. 54, no. 4, pp. 527-544, 2010.
 18. M. Seligman, K. Fall, and P. Mundur, “Alternative custodians for congestion control in delay tolerant networks,” *Proceedings of the SIGCOMM Workshop on Challenged Networks*, Montreal, Canada, 2006, pp. 229-236.



Mildred Madai Caballeros Morales

Mildred Madai Caballeros Morales is an Assistant Professor in the Computer Science at San Carlos of Guatemala University, Guatemala. She holds a Science and System Engineering degree from San Carlos of Guatemala University and a Master in Computer Engineering degree from Kyung Hee University, Korea. Her areas of research are in transportation systems, communication networks, VANETs, MANETs, Routing protocols, Clustering algorithms, context-aware solutions, and data management.



Rim Haw

Rim Haw received his B.S. and M.S. degrees from Department of Computer Engineering at Kyung Hee University, Korea, in 2008 and 2010, respectively. Since March 2010, he is working on his Ph.D. degree in the Department of Computer Engineering at Kyung Hee University, Korea. His research interests include advanced wireless network protocols, mobility management, P2P Network, and CCN.



Eung Jun Cho

Eung Jun Cho received his B.S. and M.S. degrees from Department of Computer Engineering at Kyung Hee University, Korea, in 2008 and 2010, respectively. Since March 2010, he is working on his Ph.D. degree in the Department of Computer Engineering at Kyung Hee University, Korea. His research interests include sensor network, 6LoWPAN security, Botnet detection, and IDS for WSN.



Choong Seon Hong

Choong Seon Hong received his B.S. and M.Sc. degrees in Electronic Engineering from Kyung Hee University, South Korea, in 1983, and 1985, respectively. In 1988 he joined KT, and worked on Broadband Networks as a member of the technical staff. Since September 1993, he joined Keio University, Japan, and received his Ph.D. degree in March, 1997. He had worked for the Telecommunications Network Lab, KT as a senior member of the technical staff and as a director of the networking research team until August 1999. Since September 1999, he has been working as a professor for the College of Electronics and Information, Kyung Hee University. He has served as a Program Committee Member and an Organizing Committee Member for International Conferences such as NOMS, IM, SAINT, APNOMS, ICOIN, E2EMON, CCNC, ADSN, ICPP, DIM, WISA, AsiaFI School, CFI and ICUIMC. His research interest includes network mobility, wireless networks, cognitive networks, ad-hoc networks, network security, Future Internet and network management. He is a senior member of IEEE, and member of ACM, IEICE, IPSJ, KIPS, KICS and KIISE.



Sungwon Lee

Sungwon Lee received the B.S., M.S., and Ph.D degrees in Computer Engineering from Kyung Hee University, Korea, in 1994, 1996, and 1998, respectively. From 1999-2008, he joined Samsung Electronics research and business groups on topics such as radio access network and core network development of cdma2000 1X, cdma2000 1xEVDO, WCDMA, HSPA, WiBro/Mobile-WiMAX, and IP Multimedia Subsystem (IMS). He has worked as a project leader for several trial innovative system developments. And, he was a senior engineer for system architecture, system design, and traffic engineering for several commercial product developments. He has published more than 100+ patents according to mobile broadband networks, including more than 20 registered US patents. He is currently an Associate Professor of Faculty with the Department of Computer Engineering, Kyung Hee University, Korea. His current research interests are in mobile broadband wireless networks, cellular communications, machine type communications, time synchronization protocols, wireless medium access control protocols, and mobile services. He is an Associate Editor for the Journal of Korean Institute of Information Scientists and Engineers - Computing Practice and Letters, Journal of the Korea Society of Computer and Information, and an Associate Director for the Open Standards and Internet Association.