science, engineering and health studies

Research Article

Science, Engineering and Health Studies
https://li01.tci-thaijo.org/index.php/sehs
ISSN (Online): 2630-0087

Enhanced data dissemination for vehicular content-centric networking (VCCN) using cluster maintenance system

Jirawat Thaenthong[1*] *and Sangduan Chootong*[2]

[1] College of Computing, Prince of Songkla University, Phuket Campus, Phuket 83120, Thailand
[2] Information Technology Department, Phuket Technical College, Phuket 83000, Thailand

***Corresponding author**:
Jirawat Thaenthong
jirawat.t@phuket.psu.ac.th

Received: *8 February 2021*
Revised: *22 February 2021*
Accepted: *4 April 2021*
Published: *14 December 2021*

Citation:
Thaenthong, J., and Chootong, S. (2021). Enhanced data dissemination for vehicular content-centric networking (VCCN) using cluster maintenance system. Science, Engineering and Health Studies, 15, 21040006.

ABSTRACT

Vehicular content-centric networking (VCCN) is the new architecture for vehicular networks. The nodes in the VCCN use the named data technique to route packets without using an IPv4/IPv6 address, hence avoiding IP/network mobility issues. With the limited buffer storage and the vehicle's traffic load, vehicular clustering is needed to allow only one (cluster head) node from each of the vehicular cluster cache contents. However, the VCCN incorporates dynamic topology changing, and the cluster head (CH) can leave without notification. This situation causes delivery delay and data lost access issues. This paper proposed the cluster maintenance system that allowed the secondary CH to be selected, to assist the primary CH in storing temporary data in a maintenance situation. The mechanism also included the cluster formation and cluster policy, to optimize the cache size in each node. The results of the proposed cluster maintenance mechanism in VCCN-C enhanced the performance in terms of the delivery delay, and hops count was better than the traditional VCCN. The overall average delivery delay and hop count of the proposed system, VCCN-C, were better than the original VCCN, about 50-65% and 51%, respectively. However, the signalling overhead was slightly higher than the original, about 7-9 bytes.

Keywords: vehicular content-centric networks; cluster maintenance; primary cluster head; secondary cluster head; data dissemination

1. INTRODUCTION

Each vehicle is expected to be equipped with a GPS module and multiple interfaces such as 802.11p and LTE in contemporary networks. The vehicles will perform the vehicular ad-hoc networks (VANETs) with dynamic topology changing, to allow sending/receiving data among vehicles – vehicle-to-vehicular (V2V) communications; the vehicle can connect to the internet with roadside units (RSUs). The vehicular communications will support both regular applications and emergency alert applications, i.e., an accident on the road.

Vehicular content-centric networking (VCCN) is the new architecture for vehicular networks and is based on VANET. The nodes in the VCCN can route packets between the source and destination node without using an IPv4/IPv6 address, avoiding the problem of IP/network mobility and wrong routing path (Wang & Wang, 2019). The VCCN used the named data network technique (Jacobson et al., 2012), which is composed of 3 parts: content store (CS), pending interest table (PIT) and forwarding information base (FIB). The primary process is to request and respond to the information. Grewe et al. (2018) analyzed the comparison of content-centric networking architectures that can be useful for vehicular networks; they concluded that content-centric networking with a named data technique is the most suitable approach for vehicular networks. The process of using named data technique for VCCN is demonstrated in Figure 1.

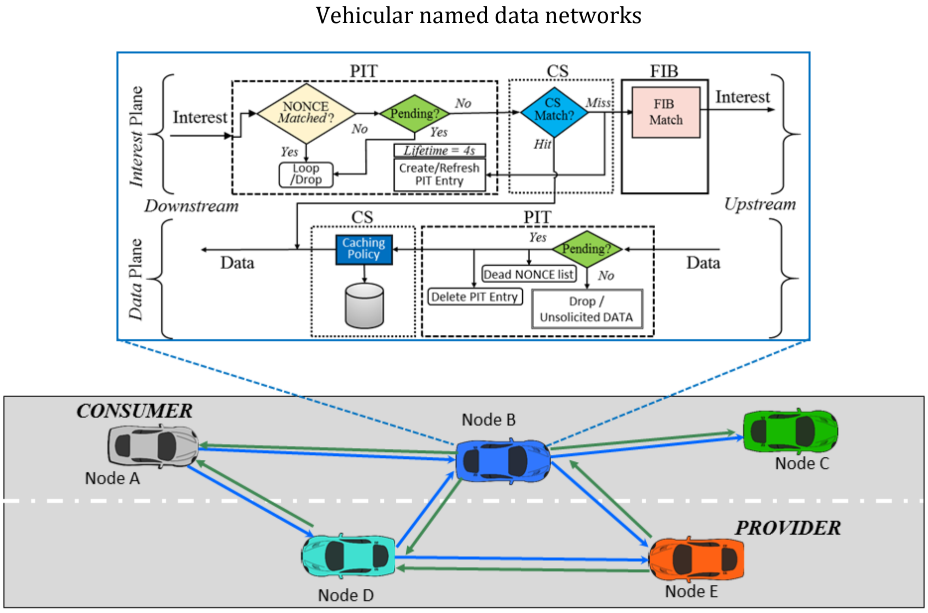

Figure 1. The process of using named data technique for vehicular content-centric networking

As seen in Figure 1, node A acts as a consumer, and node E acts as the provider. Node E already has the data node A requested. When node A initiates the interest (request) packet to ask for data, it will send a packet to its neighbours (node B and node D). These nodes receive the interest packet and then verify from the PIT table if they have the data requested or not. If they have the required data, they ignore the packet. If not, they check the current interest packet. If it exists in the waiting queue, the node deletes the interest packet. If in the case that it does not exist, the node would search for the required data in the CS and send it back to the requester node. In case the data requested do not exist in the CS, the node will forward the interest packet to neighbour nodes (node C and node E). If all nodes in the network do not have the requested data, node E will send the data back to node B and node D.

Every node that receives the data packet will check the PIT table's request to ensure the received data belongs to the interest packet. After that, the data are cached with a basic technique (e.g., LRU/LFU) (Amadeo et al., 2014; Zhang et al., 2019) in the CS as back up for future requirements.

Out of the several frameworks introduced for VCCN, Su et al. (2017) and Wang and Wang (2019) proposed one for the next generation. Su et al. (2017) put forward the content delivery exchange between vehicles based on naming information. The contents can be stored in each vehicle according to their priorities, which are determined by vehicle density and content popularity, and RSUs are designed to store contents and forward data to the requesting node (vehicle) on the road. The framework's limitation is unsupported cloud-based vehicular networks and alternative backup node on the vehicle in case the system of the RSU is down.

Wang and Wang (2019) provided the framework that supports content-centric networking, which used address-centric unicast instead of the content-centric broadcast, to achieve content acquisition. It solves content delivery because it depends on reverse paths, to return the requested contents. However, this is a trade-off for using IP-based networking inside content-centric networking. Hence, the mechanism is complicated for vehicular network implementation. Nevertheless, the VCCN can work with the new wireless communication, such as 5G and the blockchain system. Ortega et al. (2018) suggested using a blockchain system and 5G network to record transactions on vehicular communication, with high reliability and integrity while validating the information that was changed.

In VCCN, when the provider sends data, as requested from the consumer, to reduce data communication traffic, caching data on every node along the reverse path is quite helpful for data access response time. However, caching data on every vehicle requires a massive buffer on the vehicle. The policies or mechanisms were designed to help limit the buffer needed on a vehicle and improve content distribution performance. Chootong and Thaenthong (2017) proposed the caching policy with an optimization cache space technique for VCCN to reduce buffer storage on each vehicle. The simulation results show the suggested mechanism outperforms the LFU and LRU in terms of the average cache hit ratio. Yao et al. (2019) came up with a cache replacement scheme named popularity-based content caching (PopCC); it adopted the hidden Markov model (HMM) to predict the content popularity based on the characteristic of the received interests, request ratio, request frequency and content priority. The mechanism was not optimized for emergency consumption on the vehicle.

Jiang et al. (2017) proposed content clustering based on content type and a prediction-based caching strategy. The system supports distributed caching management by dividing the contents into two nodes. The concept of a backup node is helpful to design the backup system on the vehicular networks. Yao et al. (2018) introduced a cooperative caching scheme, based on mobility prediction in VCCN. The mechanisms were designed to support the caching of popular contents at a set of mobile nodes that visit at the same hot spot areas. The system was devised so that it was able to support the limited buffer on the vehicle. The mobility pattern is expected to improve the accuracy of the caching-decision mechanism. Yao et al. (2021) presented a new cooperative caching scheme based on social attributes and mobility. The system is based on the observation that vehicles move around and are liable to contact each other according to drivers' common interests or social similarities. The proposed system's lack of support minimizes energy consumption and the policy to encourage each vehicle to serve other vehicles by caching contents actively. Energy consumption is not the central issue of the vehicle's vehicular network, as it is always fully battery supported. However, Zhang et al. (2019) proposed a mobility-aware vehicular caching scheme in VCCN, to support the minimization of network energy consumption. The system is designed to optimize caching decision making with optimal energy efficiency, and the model is approved by the MATLAB model without a testbed.

Data dissemination in VCCN requires all nodes to support caching contents, to improve data access response time. One popular mechanism to reduce the number of cached nodes is called vehicular clustering. Tomar et al. (2010) suggested a concept of cluster-based data dissemination. This mechanism helps to reduce the problem of data broadcasting storms. The central node is chosen in each cluster to support caching or sending/receiving data between clusters and is responsible for forwarding data to other nodes in the cluster. Clustering reduces data spread and increases the overall data rate of vehicular communications. The vehicular clustering technique allows each vehicle in the vehicular networks to be classified with a specified vehicle speed. Farooq et al. (2016) proposed the vehicular communication framework, which included the VANET cluster scheme (VCS). The VCS consists of two processes: VANET cluster formation (VCF) and vehicle cluster maintenance (VCM). VCF supports cluster formation based on vehicle speed (three levels of speed: 60 km/h, 90 km/h and 120 km/h), and the VCM supports joining, leaving and merging the vehicle to a vehicular network. Each vehicular cluster defines a node to be a cluster head (CH) and performs the caching of data. Therefore, only one node in each cluster has the responsibility to cache content. The number of cached nodes is reduced, and the interest packets' signalling overhead between nodes decreases. However, the vehicle that acts as the CH can be left/joined during the communication, based on the dynamic topology change of the vehicular network. Finding a new CH takes a particular time and raises the problem of losing the cached data.

Several proposed techniques support using CH selection mechanism for data dissemination. Tomar et al. (2010) and Ephremides et al. (1987) suggested clustering concepts by choosing each cluster's head node. This node acted as a gateway and data transmission control of a cluster. However, Ephremides et al. (1987) proposed an excellent concept of the clustering algorithm, called the linked cluster algorithm (LCA). Each node can be either a regular node, a master node or a gateway node. By default, every node has the status of a normal node. In the CH selection process, every node broadcasts its ID. The node with the smallest ID is assigned as the CH. The CH node has two or more pieces of data passed, assumed as a gateway. The process was repeated until every node is a member of at least one cluster. The use of this mechanism is that the CH is responsible for providing information to the group members, and the gateway node supports the connection between the CH of the different groups. Therefore, this architecture can reduce overhead in the network.

Chiang et al. (1997) proposed the least cluster change (LCC) algorithm, an enhanced version of the LCA algorithm, by adding maintenance steps to reduce resource consumption in cluster reforming. For example, if a common node moves out of an existing cluster, it will not join the other clusters. It will create a new cluster and become a cluster head. The limitation of this concept is that it may produce a high number of clusters. Fan et al. (2011) suggested a dynamic clustering algorithm (DCA) that chose the CH from the cluster relation (CR); the algorithm improves the CH selection process. CR value is calculated from the total spatial dependence divided by the number of neighbour nodes.

Selvan et al. (2013) and Farooq et al. (2016) proposed a clustering technique based on vehicle speed. Selvan et al. (2013) suggested developing a clustering technique based on a vehicle's density, speed and position, to reduce the delay overhead. The proposed technique is based on vehicle density on the highway road. The speed and location of the vehicles are used for CH selection. The vehicle is selected to be the new CH if it has a speed higher than the original CH. Farooq et al. (2016) proposed a mechanism to optimize the data distribution by clustering the network into three groups based on the vehicle's speed: slow, medium and fast; the mechanism manages cluster stability. When two clusters move close with similar speed, they would be merged. The process helps to defeat the individual vehicle joining and leaving the cluster. However, both Chiang et al. (1997) and Farooq et al. (2016) used the non-optimal caching technique, such as LFU or LRU.

Morales et al. (2011), Ahizoune and Hafid (2012), Sheth and Kothadiya (2014), Singh et al. (2015) and Dwivedy et al. (2019) proposed stability clustering mechanisms. Singh et al. (2015) presented a novel algorithm to construct stable clusters, to achieve cluster-based routing. The algorithm is based on the position and direction information. The priority of a node becoming a CH is computed by the mobility information of neighbour nodes. The node with a high value of stable neighbours maintains the closer distances to its stable neighbours and has a speed close to its neighbours' average rate and would be selected to be a CH. Ahizoune and Hafid (2012) proposed a new stability-based clustering algorithm (SBCA) protocol to diminish the cluster maintenance process's overheads. The operation has two phases. First, the setup phase is the primary cluster head (PCH) selection procedure depended on the vehicle's speed. Second, the maintenance phase is secondary cluster head (SCH) selection for information backup from PCH. Sheth and Kothadiya (2014) came up with the stable clustering mechanism for the highway scenario in VANETs. This algorithm performs clustering with the direction of the vehicle and its average speed. It used the pre-specified speed range to perform the cluster of vehicles; the frequency of CH changing is decreased. As a result, the chance of having stable clusters would be generated more, compared to other algorithms. Morales et al. (2011) improved the clustering stability with the proposed adaptable mobility-aware clustering algorithm based on the destination in vehicular networks (AMACAD). The mechanism used the destination of the vehicle to reduce the number of re-clustering (Dwivedy et al., 2019). The algorithms are composed of two parts: clustering and CH selection, and the decision-making algorithm. The algorithms are designed to maximize the data dissemination efficiency and success rate. However, the algorithm is not designed for clustering maintenance, and the dynamic changing of the CH impacts the stability of vehicle clustering. Khan et al. (2019) proposed a two-level clustering scheme for efficient data dissemination in vehicular networks supported by 5G. The mechanism uses the algorithm with three factors (relative velocity, connectivity and link reliability). The level-1 CH is designed for vehicular-to-vehicular communication, and the level-2 CH is designed to reduce the gateway selection (vehicular to infrastructure communication). The proposed mechanism is helpful for 5G data communications.

This research focuses on the VCCN network. Each vehicle is a membership of a cluster. Each cluster has communication via the CH. This paper proposed the cluster maintenance system, to improve the data dissemination for VCCN; it included two main parts, as follows:

- Cluster formation and CH selection: the proposed mechanism to form vehicle clustering and to select a suitable node was a cluster leader/head. The CH acts as an access point in a wireless network, to provide the needed information to all members and allow all members to communicate with the others in another cluster.
- Caching policy: the proposed caching mechanism was applied to determine what information should be stored or replaced with new information. This research's caching policy is based on the CRCP (Chootong & Thaenthong, 2017), an optimized cache space technique.

This research's performance metrics were delivery delay, signalling overhead and hop count. The proposed system was compared with the traditional data dissemination process of VCCN.

2. MATERIALS AND METHODS

2.1 Conceptual technique

The cluster formation and cluster maintenance in vehicular networks were designed based on Farooq et al. (2016), as follows:

1) VANET cluster formation (VCF)

- Cluster speed limit (CSL) is a three-lane segmentation by speed. It is divided into three clusters: (1) clusters of the slow lane (CS), speed less than 60 km/h; (2) clusters of the medium lane (CM), the speed of 60 km/h but less than 90 km/h; (3) clusters of the fast lane (CF), the speed of 90 km/h but not more than 120 km/h.
- Cluster threshold value (CTV) is the criterion used to determine if the vehicle remains in the same speed cluster.
- CSL and CTV process clusters' construction.
- Vehicles cluster header election (VCHE) determines the leader of a cluster, using each vehicle's speed compared to CTV.

2) Vehicle cluster maintenance (VCM) consists of joining, leaving and merging. When the vehicle enters a cluster, the speed is monitored by the CSL value.

The vehicle speed was checked periodically. If the speed is still in the same speed range, the vehicle is assumed to still be in the cluster. However, if the vehicle speed changes, the CSL value checks the range of cluster. This mechanism assists in monitoring the cluster stability. The main disadvantage of the process is finding the suitable CH cause of the delay of each neighbour's speed information gathering in the cluster. This paper applied the VCF and VCM mechanism with the second CH selection in each cluster, to reduce CH selection timing when the PCH was unavailable.

The k-means are a popular clustering technique (Jain, 2010; Yuan et al., 2004) applied for vehicular cluster formation, and are able to locate the appropriate midpoint of the cluster. The processes are to 1) determine or randomize the initial value of k and define the initial center k, the point called the midpoint or centroid, and 2) take all the nodes into a group by defining each node's distance and the designated cluster center. If a node is closed to the center, it becomes a cluster member. Let $\mathrm{p_k}$ be the coordinates of calculation points, $\mathrm{q_k}$ be the coordinates of the centroid, $(\mathrm{x_1}, \mathrm{y_1})$ be coordinates of calculation points and $(\mathrm{y_1}, \mathrm{y_2})$ be coordinates of calculation points, then the distance between two nodes, dis_k^i are as the following, Equation 1 or 2.

$$dis_k^i = \sqrt{\sum_{k=1}^{n}(p_k - q_k)^2} \tag{1}$$

$$dis_k^i = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{2}$$

3) Let k be the subcluster, and dis_k^i be the average distance of each cluster. Find the mean, as per Equation 3 of each cluster, then give it a new center value.

$$dis_k^i = \frac{1}{|k|}\sum_{i=1}^{i<|k|}\sum_{j=i+1}^{j\leq|k|} \tag{3}$$

4) Repeat step 2 until the mean or the center of each cluster is not changed.

After the process, the number of k clusters and the centroid's coordinates (CH position) in each cluster were calculated. The k-means technique provides a CH position with the optimal cluster spacing.

2.2 Definition of a cluster

In this paper, a cluster is defined as follows: TR is transmission range, D_{th} is distance threshold, ΔD is a relative distance, L is cluster length, T_{Hello} is the waiting time for 'Hello' message at CM_i, ClusterID is the ID of a cluster, CM_{table} is the table of cluster member, CM_i(i=1,…,n) is a cluster member, CH is a cluster head and CH_{mer} is a CH merge.

2.3 System architecture

This research focused on cluster maintenance in the VCCN, including an SCH selection in each vehicle cluster to be a backup node for the PCH. The proposed mechanism used caching policy based on the previous work of Chootong and Thaenthong (2017). The approach is expected that all vehicle's built-in GPS devices can receive location information, speed and movement direction. Furthermore, each vehicle can calculate the speed, compared to its neighbours, and determine the distance.

The architecture of this research, as shown in Figure 2, consists of three types of nodes: ordinary node, PCH node and SCH node, as follows:

1) ordinary node is a common node without backup functions,

2) PCH node is a master node that serves its members within a cluster,

3) SCH node is a backup node that serves its members within a cluster if the PCH node moves out from the cluster.

All nodes are supported by wireless technology, clustering and GPS module. The CH is applied with CRCP caching and information collection methods proposed by Chootong and Thaenthong (2017).

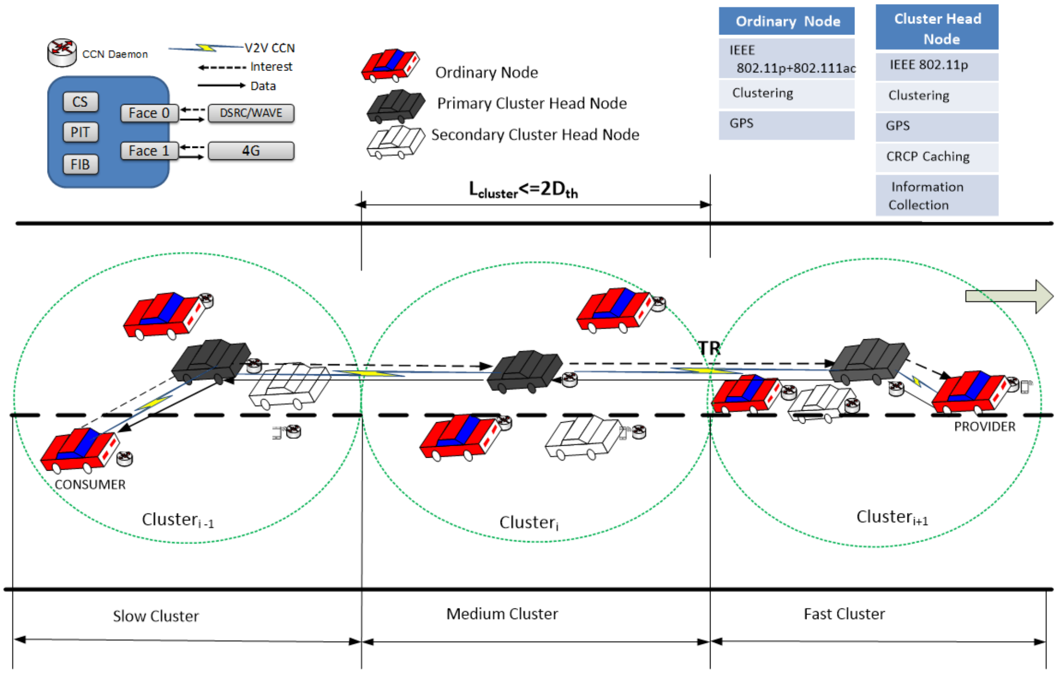

Figure 2. System architecture

2.4 Cluster formation and CH selection

The steps of cluster formation and CH selection are presented in Figure 3.

2.5 Cluster maintenance

In this paper, the cluster maintenance system was proposed to keep the ongoing vehicular communication process. The evidence of cluster maintenance was divided into 7 cases: PCH-active and SCH-passive, PCH-active and SCH-leave, PCH-leave and SCH-passive, PCH-leave and SCH-leave, joining a cluster, cluster merge and leaving a cluster. The algorithms that support in each case are explained as follows:

1) PCH-active and SCH-passive: this is the typical situation in which the PCH acts as the primary CH to support data communication for all members, while the SCH is in passive mode (backing up data and waiting for active status), as shown in Figure 3.

2) PCH-active and SCH-leave: the algorithm is used as in the PCH situation while the SCH leaves from a cluster. The process needs to discover a new SCH, as shown in Figure 3.

3) PCH-leave and SCH-passive: Algorithm 3 provides the situation when the PCH leaves a cluster and the SCH is promoted to be the new PCH. The process needs to discover a new SCH, as shown in Figure 4.

4) PCH-leave and SCH-leave: Algorithm 4 is the situation that both PCH and SCH leave from a cluster. The new PCH and SCH are processed as shown in Figure 4.

5) Joining a cluster: once a PCH_i receives a $M_{ReqJoin}$ message from vehicle V_j, it calculates its distance between PCH_i and V_j. If $\Delta D \leq D_{th}$ and the source vehicle do not have their CM, the PCH_i adds V_j as its CM and records V_j's position to its CM_{table}. The PCH_i will broadcast M_{Adv} notification messages to all CMs to update their membership. The process of joining a cluster works is shown in Figure 5 and Algorithm 5.

6) Cluster leaving: on the road, vehicles may join and leave clusters all the time. Each CH has a validation function for the presence of its CM, and each CH has a timer T_{Hello} for each T_{Hello} interval; CH checks its 'Hello'. When the CH_i receives a 'Hello' from CM_i, it checks the ΔD of CM_i in L (D_{th}) range. If $\Delta D \leq D_{th}$, CH_i sets CMi_{status} to 1; otherwise, it sets CMi_{status} to 0. After that, CH_i broadcasts a M_{Leave} message to inform CM_i that it is not a member of CM_i and deletes it from CM_{table} of CH. The process works are shown in Figure 5 and Algorithm 6.

7) Cluster merge: this situation occurs when the CH of 2 clusters move close, and these clusters have overlapping areas. Both clusters are merged into a single cluster (single CH). If the distance between 2 CHs is less than the D_{th} and both CH can hear each other within the D_{th}, CH_{i+1}. The CH that is moving behind (CH_{i+1}) will send the $M_{ReqMerge}$ message to the moving ahead CH_i. When CH_i receives $M_{ReqMerge}$ message, it will verify L. If $L \leq 2D_{th}$, a CH_{mer} will be selected as PCH.

The previous CH of two clusters (CH_i and CH_{i+1}) will send their CM list to PCH. The PCH broadcast the M_{Adv} message to inform all their CMs to change Cluster ID. The process works are shown in Algorithm 7 and Figure 6

8) Data duplication: in this process, PCH backs up data to SCH. The advantage of the process is to improve local data access and to reduce the number of forwarding requests to the neighbour cluster head when the PCH is leaving the

cluster. For every period, the relative distance between PCH and SCH is checked. If $\Delta D \leq D_{th}$, PCH will send M_{ReqDup} message to SCH, and then the PCH sends the data and status to the SCH. The process is shown in Algorithm 8.

Algorithm 1: PCH-active and SCH-passive	Algorithm 2: PCH-active and SCH-leave
1: Check status of PCH and SCH	1: Check status of PCH and SCH
2: If PCH = active and SCH = passive	2: If PCH = active and SCH = leave
3: Begin data transmission.	3: Discover new SCH
4: Data duplication process	4: Announce SCH to PCH
	5: Begin data transmission
	6: Data duplication process

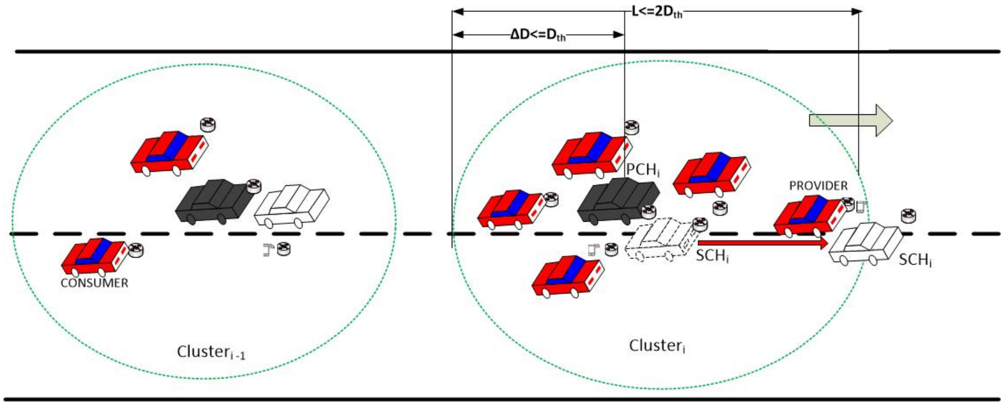

Figure 3. PCH-active and SCH-passive, PCH-active and SCH-leave

Algorithm 3: PCH-leave and SCH-passive	Algorithm 4: PCH-leave and SCH-leave
1: Check status of PCH and SCH	1: Check status of PCH and SCH
2: If PCH = leave and SCH = passive	2: If PCH = leave and SCH = leave
3: Set SCH = PSH	3: Process new PCH and SCH
4: Announce PCH to CM and NCH	4: Announce PCH to SCH, CM, and NCH
5: Begin data transmission	5: Begin data transmission
6: Process new SCH	6. Data duplication process
7: Announce SCH to PCH	
8: Data duplication process	

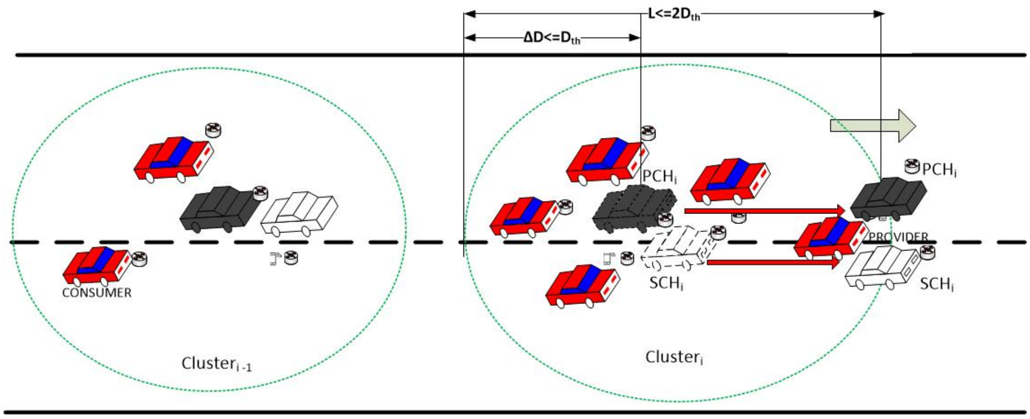

Figure 4. PCH-leave and SCH-passive, PCH-leave and SCH-leave

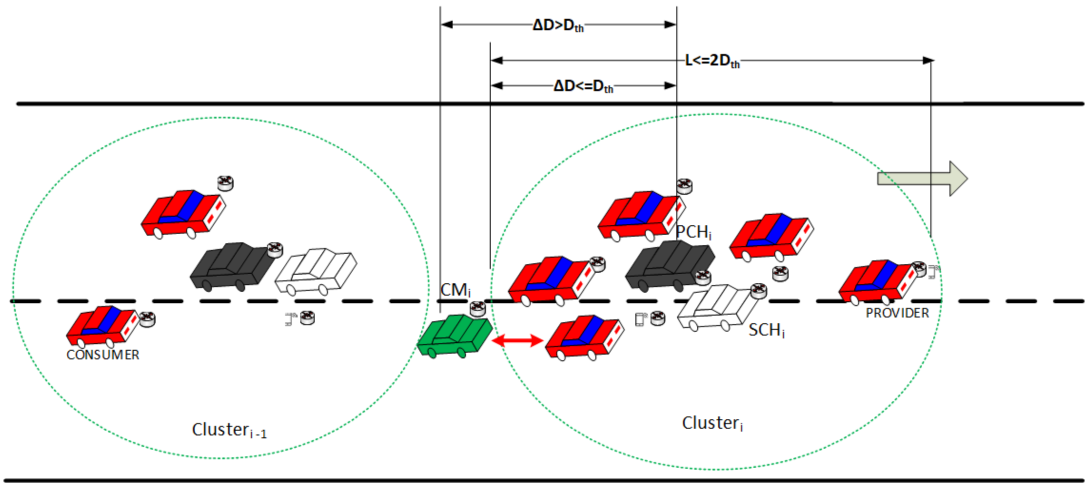

Figure 5. Joining a cluster and cluster leaving

Algorithm 5: Joining a cluster	**Algorithm 6: Cluster leaving**
1: while true do	1: while true do
2: If PCH_i receives a $M_{ReqJoin}$ from vehicle V_j then	2: CH_i receives a Hello message from CM_i
3: If ΔD between PCH_i and $V_j \leq D_{th}$ then	3: If (Check ΔD of CM_i in D_{th} range) and (ΔD ≤ D_{th}) then
4: PCH_i adds V_j to be a member	4: CMi_{status} to 1
5: Records V_j's position to CM_{table} of PCH_i	5: Else
6. PCH_i broadcast M_{Adv} to inform all CM	6. CMi_{status} to 0
7: End if	7: CH_i broadcast MLeave to inform CM_i in CM_{table}
8: End if	8: Delete CM_i from CM_{table}
9: Data duplication process	9: End if
10: End while	10: End while

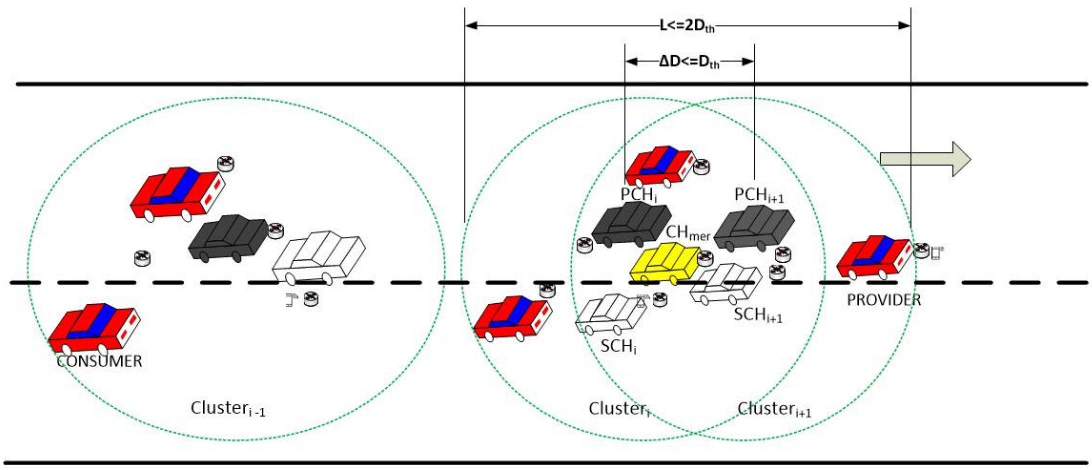

Figure 6. Cluster merge

Algorithm 7: Cluster merge	
1: While true do	10: If PCH add CMs to CM_{table} then
2: If ΔD between CH_i and CH_{i+1} ≤ D_{th}, then	11: PCH broadcast M_{Adv} to inform CMs
3: CH_i send $M_{ReqMerge}$ to CH_{i+1}	12: End if
4: End if	13: CMs change their ClusterID
5: If (CH_{i+1} get $M_{ReqMerge}$) and (L between $Cluster_i$ and $Cluster_{i+1}$ ≤ $2D_{th}$) then	14: Process new SCH
6: Select CH_{mer}	15: Announce PCH to SCH, CMs and neighbour PCHs
7: Set CH_{mer} = PCH	16: Begin data transmission
8: End if	17: End while
9: CH_i and CH_{i+1} send CMs_{List} to PCH	

Algorithm 8: Data duplication
1: While true do
2: If (ΔD between PCH and SCH) and (ΔD ≤ D_{th}) then
3: PCH broadcasts M_{ReqDup} to SCH and waits for
4: PCH forward data and state to SCH
5: Update data on SCH
6: End if
7: End while

2.6 Message type

There are six types of proposed messages; they are defined as follows: M_{Hello} includes the current speed and position of vehicles, M_{Adv} is the CH advertisement, $M_{ReqMerge}$ is the cluster merge request, $M_{ReqJoin}$ is the cluster joining request, M_{Leave} is the node leaving advertisement (CM_i's message) and M_{ReqDup} is the data duplication request (PCH's message).

Each message has a one-hop transmission, and the parameters include the following: message type, node ID, node type, cluster ID, x-coordination, y-coordination, speed, direction and the timestamp. The greyed-out field is added to the original 'hello' message. It has a total of 32 bits (4 bytes), as in Figure 7.

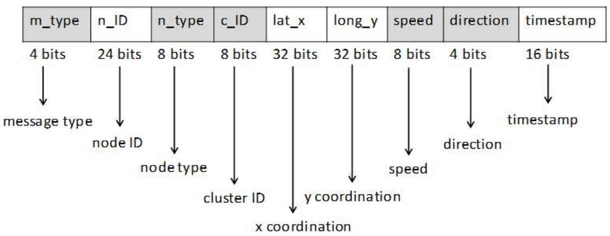

Figure 7. Hello message format

3. RESULTS

3.1 Sim simulation environment

The proposed algorithm was tested using the NDN simulation (Afanasyev et al., 2012) based on NS-3 (network simulation version 3.25), which used 802.11p technology. For each simulation, assumption parameters were assigned, such as the time for 'hello' message (T_{Hello}) was 1 s, the average speed of a vehicle was 30 to 120 km/h, TR was 250 m, D_{th} was 200 m, the data size of a packet was 1,024 bytes, the maximum data rate of a vehicle communication was 6 Mbps, the total number of nodes was 10 to 50 nodes and the number of nodes in a cluster was 3 to 21 nodes. The simulation ran in 60 s, with the frequency of data requests being 100 interests/s. The scenario had a size (1500 m x 6 m), with one-direction communication and two lanes. The simulations were tested several times with 20 different seeds.

The experiment compared the traditional VCCN and VCCN-C, with difference in traffic patterns and scenarios on the highway road. The vehicle speed was composed of three types from statistical measurement (Rudack et al., 2002; Yousefi et al., 2008). From the specific scenario, the consumer vehicle requested data from a vehicle that was far away. It was assumed that the original data was only in the provider vehicle, a data source. The exchange of data in the VCCN has several types, such as best route strategy, multicast strategy and client control strategy (Rudack et al., 2002). This research used the best route technique for data communication between nodes in the cluster.

VCCN-C is a scenario for the proposed cluster maintenance mechanism. In the cluster, it had two types of CH nodes (no gateway node used in this paper) such as PCH and SCH. This technique enhanced local data access and guaranteed the minimum latency to achieve the requested data.

3.2 Delivery delay

The simulation results were composed of the comparative analysis of traditional data communication in VCCN and the proposed mechanism of VCCN-C. The performance metrics were delivery delay and signalling overhead. There are two scenarios for simulation, VCCN and VCCN-C. Assume the consumer vehicle (sender) requests data from the provider (receiver) vehicle, and there are no data cached on every node at the beginning of simulations. The cache sizes are defined based on the CRCP reference technique (Chootong & Thaenthong, 2017)

The delivery delay calculation was calculated based on the data transmission from the producer to the consumer node. Let d_{tran} be a delay of transmission, d_{prop} be a delay of propagation, d_{prop} be a delay of process, d_{queue} be a delay of the queue and N is the node's number on a route (the provider to customer node). The delivery delay calculation for the VCCN scenario is presented in Equations (4) and (5).

$$d_{component} = d_{tran} + d_{prop} + d_{proc} + d_{queue} \tag{4}$$

$$d_{VCCN} = N * d_{component} \tag{5}$$

For the VCCN-C scenario, the vehicular network architecture was based on the cluster, and the consumer vehicle requested data from the other vehicle (provider). Assume the data exist only with the provider. Delivery delay can be calculated through Equation 6.

The delivery delay between consumer and provider vehicle within cluster C is d, which is calculated by the difference between the time of the message received (T_{Rx}) by the consumer and the message transmitted (T_{Tx}) by the provider.

$$d_{VCCN-C} = T_{Rx} - T_{Tx} \tag{6}$$

Equation 7 denotes the time interval between which the data can be sent entirely with an L bit length between the pair of vehicles *(i, j)*. It is the transmission of vehicle *i* to vehicle *j* with the data rate *f(i, j)* (Mbps).

$$d(i,j) = \frac{L}{f(i,j)} \tag{7}$$

Assume that cluster *C* consists of connected vehicles of the *h* transmission range *(h = {1,2, ..., H})*. The average delivery delay for data communication within a cluster is as per Equation 8.

$$d_{cluster} = \sum_{i,j} d(i,j) = L \sum_{i,j} \frac{L}{f(i,j)} \tag{8}$$

where *d(i,j)* is a delay of the data distribution, while vehicle *i* is connected to vehicle *j*. Assume the data rate is constant (*f = f ((i, j))* for each connection *(i, j)* in the cluster *C*, as the following Equation 9.

$$d'(i,j) = \frac{L * h}{f} \tag{9}$$

From the equation above, the final delivery delay for V2V communications is calculated by the following Equation 10.

$$d'_{VCCN-C} = d'(i,j) = \Delta T \tag{10}$$

where *d' (i, j)* is a delivery delay of the cluster's data distribution, as presented in Equation 9, and *ΔT* is the smallest interval in which each pair vehicle communicated at constant velocity *c* m/s and separated at a distance *ΔX* m.

$$\Delta T = \frac{\Delta X}{c} \tag{11}$$

3.3 Signalling overhead

The signalling overheads occurred in the service provider and consumer communication. In this paper, the proposed mechanism allowed only the consumer to send the CH's request. According to different situations, signalling overhead analysis was calculated in one request per one data packet. N_i is the intermediate node and N_{cl} is the number of clusters in the system; the analysis is divided into seven sub-situations.

1) Original VCCN: the signalling overhead in the original VCCN communication mechanism is caused when a request is sent from the consumer to all neighbour nodes and the neighbour nodes forward the message if they have no data as requested. The process is repeated until the data are sent back to the consumer. The signalling overhead can be calculated through Equation 12.

$$SH_{VCCN} = \sum_{i=1}^{i=n} (M_{interest} * N_i) \tag{12}$$

2) PCH-active and SCH-passive: this is the first case of the proposed cluster maintenance mechanism. The signalling overhead concerns the request message sent from a consumer to PCH and the forward message from PCH to neighbour PCH; it continues until a response to the consumer's data is made and a message to request backing up data at SCH is received. The Equation 13 calculates this as follows:

$$SH_{case1} = \sum_{i=1}^{i=n} (M_{interest} * N_i) + N_{cl} * M_{ReqDup} \tag{13}$$

3) PCH-leave and SCH-passive: the second case of the proposed cluster maintenance mechanism is when the PCH leaves the cluster and the SCH stands by in the cluster. This signalling overhead is caused by a consumer's request to the data provider (via the PCH) which is combined with the message that SCN is representative to act as a PCH. After that, the new PCH sends a message to inform SCH to copy the selected SCH data (new SCH). The equation is as follows:

$$SH_{case2} = \sum_{i=1}^{i=n} (M_{interest} * N_i) + N_{cl} * M_{Adv} + N_{cl} * M_{ReqDup} \tag{14}$$

4) PCH-active and SCH-leave: the third case of the proposed cluster maintenance mechanism is when the PCH is active and the SCH leaves the cluster. The signalling overhead is caused by a consumer request to the data provider, combined with the message that the new node announces to be a new SCH. Furthermore, the PCH sends a message to inform SCH to back up data. The equation is as follows:

$$SH_{case3} = \sum_{i=1}^{i=n} (M_{interest} * N_i) + N_{cl} * M_{Adv} + N_{cl} * M_{ReqDup} \quad (15)$$

5) PCH-leave and SCH-leave: the fourth case of the proposed cluster maintenance mechanism is when the PCH and SCH leave the cluster. The signalling overhead is caused by the selection process of a new PCH and SCH and the members' announcement message. Furthermore, the overhead includes the consumer's requests to the data provider and all messages generated by the PCH to inform SCH, to back up data. The equation is presented as follows:

$$SH_{case4} = N_{cl} * M_{Hello} + 2 * N_{cl} * M_{Adv} + \sum_{i=1}^{i=n} (M_{interest} * N_i) + N_{cl} * M_{ReqDup} \quad (16)$$

6) Joining a cluster: it is the fifth case of the proposed cluster maintenance mechanism. When the new node requires to join a new cluster, the signalling overhead occurs. The signalling overhead is calculated, including a message to inform the cluster. The equation is presented as follows:

$$SH_{case5} = N_{cl} * M_{ReqJoin} + 2 * N_{cl} * M_{Adv} + \sum_{i=1}^{i=n} (M_{interest} * N_i) \quad (17)$$

7) Cluster leaving: this signalling overhead is caused by a message that the PCH inform all members that the leaving node is no longer a member. The equation is presented as the following:

$$SH_{case6} = N_{cl} * M_{Leave} + \sum_{i=1}^{i=n} (M_{interest} * N_i) \quad (18)$$

8) Cluster merge: this is the sixth case of the proposed cluster maintenance mechanism. The signalling overhead is caused by sending messages to the cluster merge, including a message to inform all memberships of nodes assigned to be the new PCH and SCH and a message to inform backup data from PCH to SCH. The equation is calculated as follows:

$$SH_{case7} = N_{cl} * M_{ReqMerge} + 2 * N_{cl} * M_{Adv} + N_{cl} * M_{ReqDup} + \sum_{i=1}^{i=n} (M_{interest} * N_i) \quad (19)$$

4. DISCUSSION

The results demonstrated the comparison of the traditional VCCN and the proposed cluster maintenance for VCCN-C, based on the simulation, as follows: 1) VCCN as a traditional data delivery scenario, 2) VCCN_C1 as a VCCN-C cluster delivery scenario in the case of PCH-active and SCH-passive, 3) VCCN_C2 as a VCCN-C cluster delivery scenario in the case of PCH-leave and SCH-passive, 4) VCCN_C3 as a VCCN-C cluster delivery scenario in the case of PCH-active and SCH-leave, 5) VCCN_C4 as a VCCN-C clustering scenario in the case of PCH-leave and SCH-leave, 6) VCCN_C5 as a VCCN-C clustering scenario in the case of cluster merge, 7) VCCN_C6 as a VCCN-C clustering scenario in the case of cluster leaving. This paper has experimented with the data communications between the provider and consumer in three cases, one provider to one consumer (1-1), one provider to many consumers in the same cluster (1-M), and one provider to many consumers in each distributed cluster.

The situation environment delivered data from the provider to consumers 1 to 1 and 1 to many. Here, consumers were in the same cluster, which was distributed in the highway, and the number of consumers started from 2 nodes to 13 nodes, with background traffic of 8 megabytes per second. When the number of nodes in the network increased from 10 to 50 nodes, the delivery delay was less than the original VCCN in almost all scenarios, as shown in Figures 8-10. Figure 8 showed that the average delivery delay of all cases of the proposed VCCN-C, with one provider to one consumer data communication, had the average delivery delay lower than the original VCCN, about 82%, 48%, 57%, 82%, 48% and 80% at 50 nodes in the network, respectively. The overall average of all cases was lower than the original VCCN, which was about 65%.

Figure 9 shows the average delivery delay in case of data communication from one provider to multiple consumers in the same cluster. The proposed VCCN-C for all cases provided a delay lower than the original VCCN, as the same as Figure 8. However, increasing the number of consumers in the same cluster impacted the average delivery delay, compared to Figure 8. For example, at 50 nodes, the average delivery delay of all cases was reduced by 57%, 54%, 51%, 34%, 42% and 71%, compared to the original VCCN. The overall average delivery of all cases provided a delay lower than the original VCCN by 51%.

This paper considered only the signalling overhead when the provider's data delivery to the consumer was one to one communication with 8 Mbps for background traffic. Figure 10 shows the average signalling overhead of all cases in the situation, where nodes were incremented from 10 to 20, 30, 40 and 50 for the data delivery from a provider to consumer. When the total nodes were 20, the average signalling overhead of the proposed VCCN-C increased by approximately 7 bytes than the original VCCN. The signalling overhead is subjected. However, if multiple data packets are transferred between the provider and consumer, it does not affect the overall delivery delay.

The number of nodes/clusters impacted the average hop count in vehicular communications. Figure 11 shows that increasing the number of nodes/clusters increased the average hop count. For example, when the number of nodes is 21 nodes with the different data requested, the average hop count of the proposed VCCN-C decreased by 51% from the original VCCN.

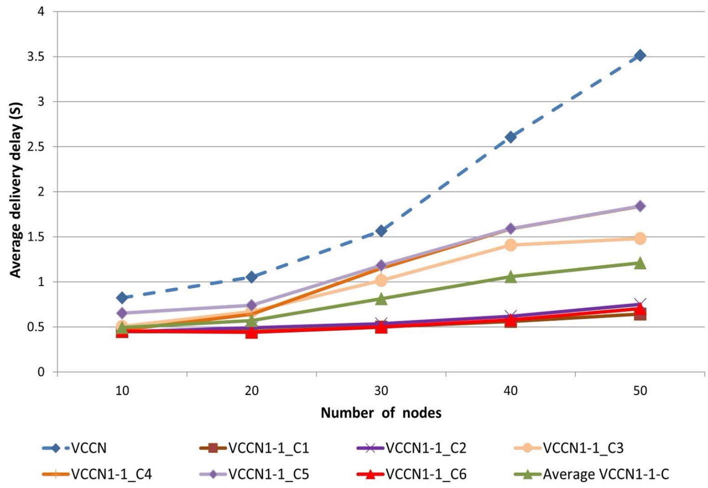

Figure 8. Average delivery delay and number of nodes: one provider to one consumer

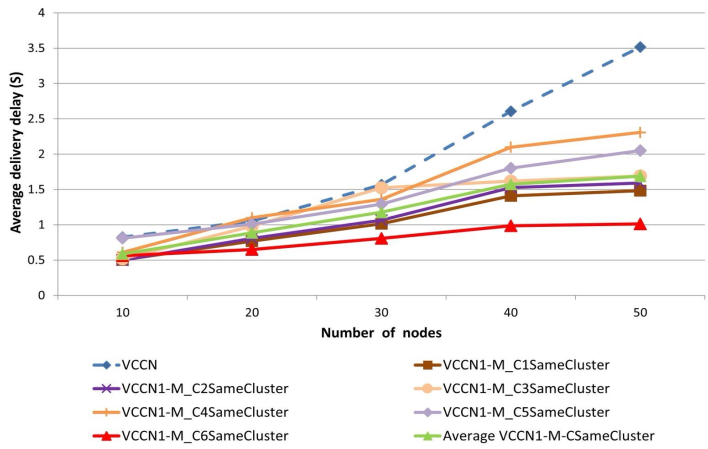

Figure 9. Average delivery delay and number of nodes: one provider to different numbers of consumers in the same cluster

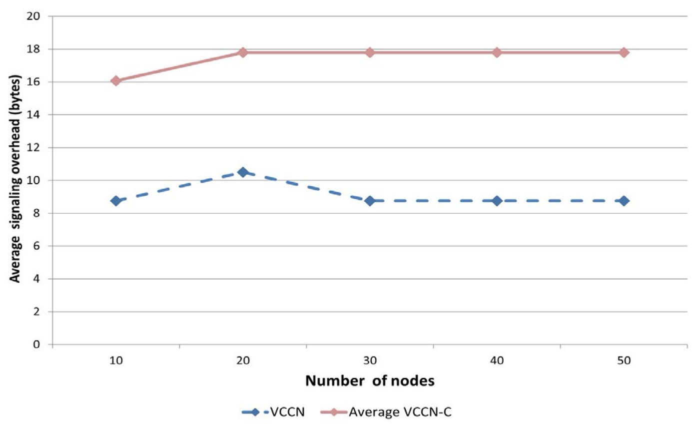

Figure 10. Average signalling overhead and number of nodes

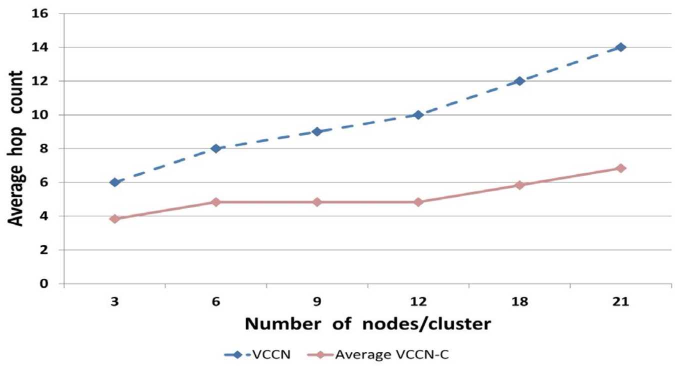

Figure 11. Average hop count and number of nodes/cluster case: different data

From the results, the proposed VCCN-C performed better than the original VCCN in terms of delivery delay and hop count for all cases (scenarios): one-to-one communication and one-to-multiple communications between the provider and the consumer. The signalling overhead is an issue because the proposed system's overhead was slightly higher than the original. However, the signalling overhead can be a trade-off; it depended on the number of data delivery between the provider and the consumer.

5. CONCLUSION

In this paper, we proposed the cluster maintenance mechanism using the cluster formation. PCH and SCH selection included the proposed caching mechanism to optimize the cache space, where the critical data should be stored in each CH – based on the CRCP. The PCH served as the primary data provider to the cluster members, and the SCH served as a backup CH node in case the PCH left the group. Additionally, we offerred a cluster stability concept allowing members within a cluster to access information faster. The simulation and mathematical analysis results were performed with the three performance indicators: delivery delay, signalling overhead and hop count. The study of the average of all metrics were compared to the original VCCN, and the proposed VCCN-C delivery concept optimized the delivery delay and hop count. The signalling overhead slightly increased due to data communication in the cluster stabilization process. However, the signalling overhead, as opposed to the number of packets received from communications, is minimal. In future work, the delivery delay between the PCH and the SCH will be optimized and the roadside unit will be used an assistive unit for the cluster maintenance system.

ACKNOWLEDGMENT

This research was supported by the College of Computing, Prince of Songkla University, Phuket Campus, Thailand.

REFERENCES

Afanasyev, A., Moiseenko, I., and Zhang, L. (2012). *ndnSIM: NDN simulator for NS-3* (NDN, Technical Report NDN-0005). University of California. USA.

Ahizoune, A., and Hafid, A. (2012). A new stability based clustering algorithm (SBCA) for VANETs. In *Proceeding of the 37th IEEE Annual Conference on Local Computer Networks Workshops*, pp. 843-847. Clearwater, USA.

Amadeo, M., Campolo, C., Molinaro, A., and Ruggeri, G. (2014). Content-centric wireless networking: A survey. *Computer Networks*, 72, 1-13.

Chiang, C. C., Wu, H. K., Liu, W., and Gerla, M. (1997). Routing in clustered multihop, mobile wireless networks with fading channel. In *Proceedings of the IEEE SICON*, pp. 197-211. New York, USA.

Chootong, S., and Thaenthong, J. (2017). Cache replacement mechanism with content popularity for vehicular content-centric networks (VCCN). In *Proceedings of the 14th International Joint Conference on Computer Science and Software Engineering,* pp. 1-6. Nakhon Si Thammarat, Thailand.

Dwivedy, B., Bhola, A. K., and Yadav, S. (2019). Cluster based multi hop data dissemination protocol in V2V networks using whale optimization technique. In *Proceeding of the International Conference on Automation, Computational and Technology Management*, pp. 228-231. London, UK.

Ephremides, A., Wieselthier, J. E., and Baker, D. J. (1987). A design concept for reliable mobile radio networks with frequency hopping signaling. *Proceedings of the IEEE*, 75(1), 56-73.

Fan, W., Shi, Y., Chen, S., and Zou, L. (2011). A mobility metrics based dynamic clustering algorithm for VANETs. In *Proceeding of the IET International Conference on Communication Technology and Application,* pp. 752-756. Beijing, China.

Farooq, W., Ali Khan, M., and Rehman, S. (2016). A novel real time framework for cluster based multicast communication in vehicular ad hoc networks. *International Journal of Distributed Sensor Networks*, 2016(12), 1-18.

Grewe, D., Wagner, M., and Frey, H. (2018). A domain-specific comparison of information-centric networking architectures for connected vehicles. *IEEE Communications Surveys and Tutorials*, 20(3), 2372-2388.

Jacobson, V., Smetters, D. K., Thornton, J. D., Plass, M. F., Briggs, N. H., and Braynard, R. L. (2012). Networking named content. *Communications of the ACM,* 55(1), 117-124.

Jain, A. K. (2010). Data clustering: 50 years beyond K-means. *Pattern Recognition Letters*, 31(8), 651-666.

Jiang, X., Zhang, T., and Zeng, Z. (2017). Content clustering and popularity prediction based caching strategy in content centric networking. In *Proceedings of the 85th IEEE*

Vehicular Technology Conference, pp. 1-5. Sydney, Australia.

Khan, Z., Fan, P., Abbas, F., Chen, H., and Fang, S. (2019). Two-level cluster based routing scheme for 5G V2X communication. *IEEE Access*, 7, 16194-16205.

Morales, M. M. C., Hong, C. S., and Bang, Y. (2011). An adaptable mobility-aware clustering algorithm in vehicular networks. In *Proceeding of the 13th International Conference on Asia-Pacific Network Operations and Management Symposium,* pp. 1-6. Taipei, Taiwan.

Ortega, V., Bouchmal, F., and Monserrat, J. F. (2018). Trusted 5G vehicular networks: Blockchains and content-centric networking. *IEEE Vehicular Technology Magazine*, *13*(2), 121-127.

Rudack, M., Meincke, M., and Lott, M. (2002). On the dynamics of ad hoc networks for inter vehicle communications (IVC). In *Proceedings of the International Conference on Wireless Networks,* Paper-id: 1051WN. London, UK.

Selvan, M., T., M., Maheshwari, Mary, S., and R. (2013). A cluster-based highway vehicle communication in VANET. In *Proceedings of the National Conference on Recent Trends in Computer Applications,* pp. 1-5. Chennai, Tamil Nadu, India.

Singh, A., and Kaur, M. (2015). A novel clustering scheme in vehicular ad hoc network. *International Journal of Applied Information Systems*, 10(3), 1-5.

Sheth, P., and Kothadiya, H. (2014). Multi-hop stable clustering in VANET on highways. *International Journal of Advance Research in Computer Science and Management Studies*, 2(5), 52-56.

Su, Z., Hui, Y., and Yang, Q. (2017). The next generation vehicular networks: A content-centric framework. *IEEE Wireless Communications*, 24(1), 60-66.

Tomar, P., Chaurasia, B. K., and Tomar, G. S. (2010). State of the art of data dissemination in VANETs. *International Journal of Computer Theory and Engineering,* 2(6), 957-962.

Wang, X., and Wang, X. (2019). Vehicular content-centric networking framework. *IEEE Systems Journal*, 13(1), 519-529.

Yao, L., Chen, A., Deng, J., Wang, J., and Wu, G. (2018). A cooperative caching scheme based on mobility prediction in vehicular content centric networks. *IEEE Transactions on Vehicular Technology*, 67(6), 5435-5444.

Yao, L., Wang, Y., Wang, X., and Wu, G. (2021). Cooperative caching in vehicular content centric network based on social attributes and mobility. *IEEE Transactions on Mobile Computing*, 20(2), 391-402.

Yao, L., Wang, Y., Xia, X., and Xu, R. (2019). Popularity prediction caching using hidden markov model for vehicular content centric networks. In *Proceedings of the twenty IEEE International Conference on Mobile Data Management*, pp. 533-538, Hong Kong, China.

Yousefi, S., Altman, E., El-Azouzi, R., and Fathy, M. (2008). Analytical model for connectivity in vehicular ad hoc networks. *IEEE Transactions on Vehicular Technology*, 57(6), 3341-3356.

Yuan, F., Meng, Z.-H., Zhang, H.-X., and Dong, C.-R. (2004). A new algorithm to get the initial centroids. In *Proceedings of the International Conference on Machine Learning and Cybernetics*, pp. 1191-1193. Shanghai, China.

Zhang, Y., Li, C., Luan, T. H., Fu, Y., Shi, W., and Zhu, L. (2019). A mobility-aware vehicular caching scheme in content centric networks: Model and optimization. *IEEE Transactions on Vehicular Technology*, 68(4), 3100-3112.