國立交通大學

資訊管理研究所

博士論文

一個在 VANETs 中取得即時交通資訊的傳輸架構

A Novel Transmission Architecture for Real-Time Traffic Information Acquisition in VANETs

研究生：張栩嘉

指導教授：羅濟群 教授

中華民國 一百零一 年 七 月

一個在 VANETs 中取得即時交通資訊的傳輸架構
A Novel Transmission Architecture for Real-Time Traffic Information Acquisition in VANETs

研 究 生：張 栩 嘉　　Student: Hsu-Chia Chang

指導教授：羅 濟 群　　Advisor: Chi-Chun Lo

國 立 交 通 大 學
資訊管理研究所

博士論文

A Dissertation

Submitted to Institute of Information Management

College of Management

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

in

Information Management

July 2012

Hsinchu, Taiwan, the Republic of China

中華民國　一百零一 年 七 月

一個在 VANETs 中取得即時交通資訊的傳輸架構

研究生：張栩嘉　　　　　　　　　　指導教授：羅濟群 老師

國立交通大學資訊管理研究所

摘要

近年來智慧型運輸系統(Intelligent Transportation System, ITS)的發展已越來越普遍。車載隨意網路(Vehicular Ad-hoc Networks, VANETs)逐漸成為 ITS 應用的優選。VANETs 的通訊模式可分為:車輛與車輛間通訊(Vehicle-to-Vehicle,V2V) 和車輛與路邊裝置間通訊 (Vehicle-to-Infrastructure, V2I)。V2V 網路具有車輛移動迅速，網路拓樸快速改變，車間通訊範圍限制之特性，使得在動態的 VANET 環境下要能相互傳送即時的交通資訊將倚賴車輛間彼此的通訊連線。本研究提出一個在 VANETs 中能充分有效地傳送即時交通資訊的傳輸架構,此傳輸架構由三種模式構成。第一個模式操作在 V2V 通訊環境下，車間的通訊距離在傳輸範圍內時,將叢集為一群具共識的群集以利緊急事故訊息能快速有效地散播，改善廣播風暴問題;第二個模式則使用 V2I 通訊的路邊裝置(Roadside Units, RSUs),蒐集行動網路手機通訊中的訊號強度進行即時交通車速之估計,並置於雲端供用路人查詢; 第三個模式是一個混合模式，V2V+V2I，它解決 V2V 車輛間資料傳送的範圍限制，協同 V2I 模式建構傳輸即時交通資訊的連線。透過模擬與現有的廣播方法比較，實驗結果顯示本研究提出的 V2V 模式和混合模式將可降低訊息傳送數量與延遲時間，從而提升即時交通資訊的傳送效率和可靠度。此即時交通資訊的傳輸架構將有助於 ITS 在車載通訊上取得與傳送即時交通資訊之運用。

關鍵字:車載隨意網路、資料散播、叢集化、行動網路、雲端運算

A Novel Transmission Architecture for Real-Time Traffic Information Acquisition in VANETs

Student: Hsu-Chia Chang Advisor: Dr. Chi-Chun Lo

Institute of Information Management

Nation Chiao Tung University

Abstract

In recent years, the development of intelligent transportation system (ITS) has become more and more popular. The vehicular ad-hoc networks (VANETs) are emerging as the preferred network design for ITS. The VANETs are based on short-range wireless communication between vehicles, which can be classified into two communication modes, vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I). V2V refers to the direct or multi-hop communications among vehicles. V2I refers to the communication with the infrastructure of roadside units (RSUs). V2V embrace high mobility of vehicles, rapid changing topology due to uneven distribution of vehicles and inter-vehicle transmission range. Therefore, in a dynamic VANET environment, the vehicle transmission's reliability will affect the real-time traffic information dissemination in a consensual and effective way. In this thesis, we propose a transmission architecture in which real-time traffic information can be disseminated both adequately and effectively. There are three modes of operation in the proposed architecture. The first mode is operated under V2V where the vehicles are within the transmission range and proposed a consensus-based cluster protocol to disseminate real-time safety alert message to cluster members. The second mode is operated under V2I; the design of RSUs is responsible for obtaining cellular-network-based data to estimate speed information on cloud. The third mode, which is a hybrid approach, V2V+V2I, is operated via the coordination between V2I and V2V in disconnected condition, particularly. In accordance with the simulation results, we notice that, the performances of the proposed V2V mode and hybrid mode can downgrade the number of message dissemination and improve the broadcast storm problem. Therefore, the novel transmission architecture would be useful for acquiring and transmitting real-time traffic information between vehicular communications for ITS.

Keywords: *Vehicular Ad-hoc Networks (VANETs), Data Dissemination, Clustering, Cellular network, Cloud Computing.*

誌謝

本論文得以完成，我衷心地要感謝許多人給予我的鼓勵、支持和機會!

首先感謝我的指導教授–羅濟群老師於博士班的學習過程中悉心的指導並支持我的研究主題，給予我充足的資源和明確的方向。期間得以窺見師者風範，習得宏觀的思維、嚴謹的研究態度和樂觀正向的生活哲學，實則為更大的收穫。同時感謝口試委員劉敦仁教授，楊建民教授，周世傑教授和陳昱仁教授於口試期間費心指導，提供寶貴的建議，讓本篇論文內容得以更臻完善。

感謝最疼愛我與我最愛的家人，謝謝父母全力的支持，讓我無後顧之憂地完成學業；謝謝仁豪貼心的相伴和鼓勵，總是為我分憂和設想，為我們的生活增添歡樂和活力。

感謝 Netlab 的各位夥伴們。感謝俊傑學長、邦曄學長和婉佳學姊對後進的照顧；感謝泊寰學長、斯寅學長和志華學弟研究協助和合作；感謝博士班同窗:意鈞、輝哥和劉蓉的相互扶持；感謝歷屆的學弟妹:強哥、小牛、Andy、媛如等於研究過程的支援；以及感謝其他未提及的實驗室成員。感謝淑惠在所務上的幫忙和照顧；感謝交大土地公保佑我在新竹一切平安。

謹獻上最真誠的謝意，並與您們一同分享這個喜悅。

謝謝大家!

Table of Contents

Figure Index

Table Index

Pesudo Code Index

Chapter 1 Introduction

1.1 Research Background and Motivation

Intelligent Transportation System (ITS) has been developed to monitor traffic information, to reduce car accident, and to improve their comfort degree [1]. Vehicular ad-hoc networks (VANETs) are emerging as the preferred network environment with the objectives of improving the road safety and traffic efficiency for ITS. VANETs are networks in which each node is a vehicle. Such systems aim to provide communications between individual vehicles and between vehicles and nearby fixed equipment, or roadside units. Vehicles can communicate with nearby vehicles known as a V2V communication and also with road side infrastructure known as V2I [2]. Features of VANET are high mobility of vehicles, vehicles move on predefined roads, no power constrains, and rapid topology changes. The goal of VANETs, and more broadly vehicular networks, is to improve traffic safety by providing timely information to drivers and concerned authorities. The VANETs technologies are based on short-range wireless communication between vehicles. The allocation of 75 MHz in the 5.9 GHz band that is licensed for dedicated short range communications (DSRC), which aimed at enhancing bandwidth and reducing latency for V2V and V2I communication[3]. Also, the authors analyzed the pros and cons of V2Vand V2I [4]. A hybrid approach is necessary to enhance the development of ITS.

In VANETs, broadcast is typically used to disseminate traffic-related information within a certain area. The roadside unit that broadcasts traffic information should periodically rebroadcast the message to keep it alive for as long as needed. As a result, a broadcast storm may arise if the traffic density on the road and the frequency at which the RSU broadcasts the message are high. The direct impact of a broadcast storm in this case is waste of processing time and bandwidth, and increased medium access delay. However, a more serious impact of the broadcast storm is safety-related service disruption. For example, other urgent safety messages might get lost or delayed during a broadcast storm.

Data dissemination approach in VANETs are refers multi-hop communications among vehicles or store-carry-forward message in order to preserve the network connectivity [5],[6]. In push based data dissemination, the data can be efficiently delivered from moving vehicles or fixed base station (RSU) to another vehicles. Pull based data dissemination is the type where any vehicle is enabled to query information about specific location or target. This is one form of request and response model. The V2V dissemination is flooding. The flooding approach is good for delay sensitive application and also suitable for sparse networks during low traffic conditions. In this paper, focus on well-connected network, we consider the broadcast storm problem on major highways. The aspiration of our research is to propagate adequate and real-time traffic information to users in VANETs. The objectives of the proposed transmission architecture are as follow:

1. Disseminate real-time traffic information for safety alert with connected transmission range in V2V;

2. Consider the V2I schema, by integrating heterogeneous wireless technologies, obtaining cellular-network-based data [7] to estimated traffic speed for effective V2I communications;

3. Improve broadcast storm problem with well connected transmission in hybrid communications, V2V+V2I.

1.2 Proposal Outline

The remainder of this thesis proposal is structured as fellows. Chapter 2 contains the literature reviews which include the descriptions of VANETs, consensus-based election algorithms, cloud computing and its related technologies. Chapter 3 will depict the first mode of the proposed V2V communications model, which is operated with a design of consensus-based cluster algorithm to propagate messages. Chapter 4 will describe the second mode of the proposed V2I communications model, which adopts fingerprint position algorithm from the RSUs to estimate traffic speed based on cellular network data. Chapter 5 will present the third proposed hybrid communications model, V2V+V2I. The simulation results and analyses obtained in the message dissemination tests will be shown in Chapter 6. Finally, the conclusion and future works are presented in Chapter 7.

Chapter 2 Literature Review

VANETs are emerging as the preferred environment of propagating the road safety and real-time traffic data for ITS. In this chapter, several literatures related to VANETs communication technology, distributed algorithms, estimating traffic information from cellular network, and data processing based on cloud computing will be reviewed and discussed briefly.

2.1 Vehicular Ad Hoc Networks (VANETs)

VANET is a set of vehicles that communicate via short-range wireless technologies such as IEEE 802.11 and DSRC. Each vehicle participating in the VANET periodically produces reports regarding the traffic condition it is experiencing.

2.1.1 Broadcasting Algorithms

Several vehicular broadcasting algorithms in VANETs are in the following [8],[9],[10],[11],[12]:

(1) Simple broadcast

This is the simplest protocol used in V2V message dissemination for VANET. When there is an accident, safety alert message will be sent to all vehicles approaching towards accident site. When a vehicle receives a broadcast message for the first time, it retransmits the message. The vehicle then ignores all subsequent broadcast messages (with same ID) it receives, from other vehicles rebroadcasting the same message. There are two main problems in this simple broadcast method. First, there are a lot of redundant rebroadcast messages because of flooding. Thus, when a message reaches n hosts for the first time, n replications will be sent. Second, there is a high probability that a message will be received by many hosts located in a close proximity. Every host will severely contend with one another for access to the medium, known as the "broadcast storming".

(2) *P*-persistence

This method tries to reduce broadcast storm problem by using a stochastic selection method to decide the vehicle that will rebroadcast the alert message. When a vehicle receives a broadcast message for the first time, the vehicle will rebroadcast the alert message with a random probability *p* (Figure 1). This method will help to reduce the number of rebroadcasting vehicles and thereby broadcast storming problem. However, all nodes that receive broadcast message decide not to rebroadcast which will cause the loss of alert messages.

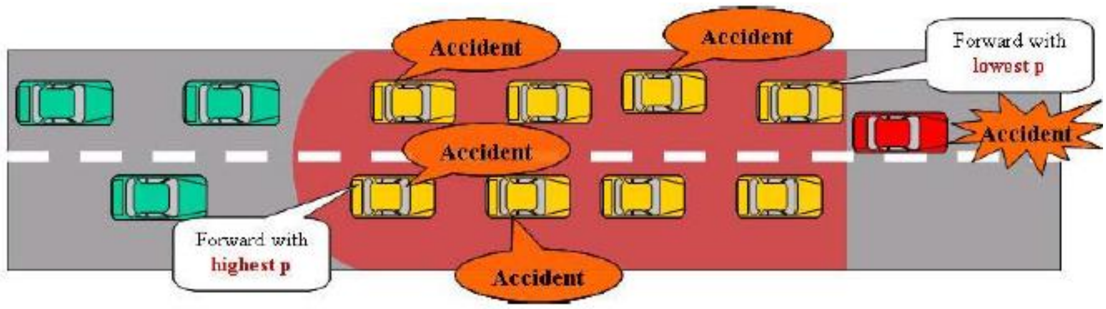

Figure 1: *P*-persistence broadcast [9]

(3) Slotted p-persistence

This is an improvement over *p*-persistence protocol. Upon receiving a packet, a node checks the packet ID and rebroadcasts with a pre-determined probability *p* at the assigned time slot Ts_{ij}, if it receives the packet for the first time and has not received any duplicates before its assigned wait time slot expires. Otherwise, it discards the packet. As shown in Figure 2, each node in this scheme also buffers the message for a certain period of time T ([Ns - 1] × WAIT TIMEE + δ ms), and retransmits with probability 1 to prevent message “die out”. The performance of the slotted p-persistence depends on the value chosen for the re-forwarding probability *p*.

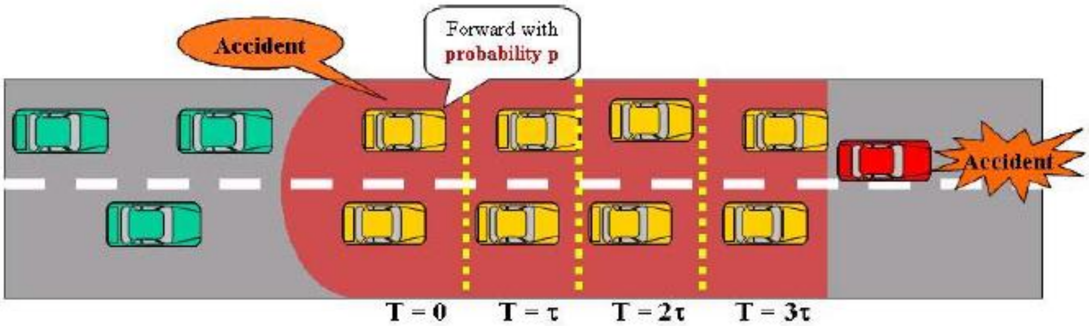

Figure 2: Slotted p-persistence broadcast [9]

(4) TLO [11]

When there is an accident, the victimized vehicle broadcast an alert message. Vehicles that receive the alert message do not rebroadcast it immediately. They will use TLO (The last one) algorithm to find the last vehicle i.e. the one furthest from the place of accident. That particular vehicle will rebroadcast the message while the other vehicles will wait for a threshold time interval to take a decision about rebroadcast. Only the node, which is designated by TLO as the furthest node (and all nodes are aware of it), will rebroadcast the alert message. When the threshold waiting time interval expires, and other nodes do not receive the same alert message again (which is supposed to come from the furthest node), there is a problem in rebroadcasting.

TLO runs again to find the next candidate as last node, which then broadcast the alert message. This is repeated until a successful rebroadcast is done. As shown in Figure 3, vehicle A receives an accident event and sends alert messages to the following vehicles which are in alert message range. They will start TLO algorithm to choose the furthest vehicle which would rebroadcast it. In this case the vehicle B, E and G (in order) are chosen to rebroadcast alert messages to the following nodes.

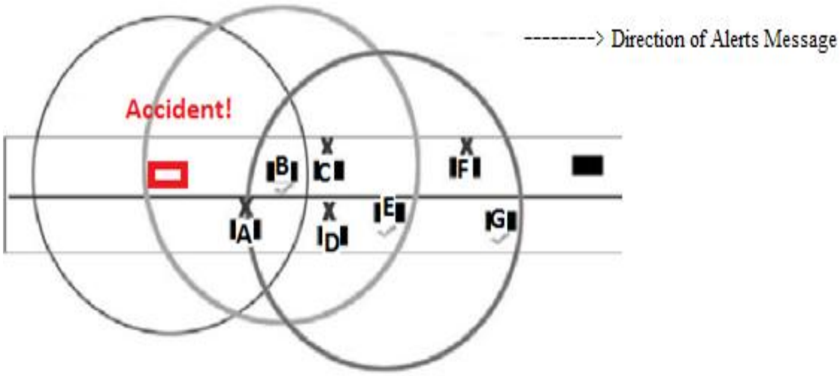

Figure 3: TLO broadcast [11]

(5) APAL [12]

Adaptive Probability Alert Protocol (APAL) does not need location information of the vehicle. The probability of alert message broadcast is adaptively set such that broadcast problem is minimized. There are 4 steps in APAL to let the received node to retransmit the message adaptively.

Step 1: If a node receives an alert message for the first time, it will wait for a random time interval Δ_{t1} which is decided with uniform random probability from a fixed interval which is set depending on traffic density.

Step 2: Every time an interval for a vehicle expires, in case the vehicle receives duplicate alert message during this Δ_{t1} time interval, i.e., one or more of its neighbors have already rebroadcasted it, the vehicle will refrain from broadcasting it again.

Step 3: When the interval expires and the node does not receive any duplicate message, the vehicle will rebroadcast it with a high probability *Pi*.

Step 4: Set a life time number and maximum number of message duplicated number for message handler. The adaptive probability and interval is adopted to actuate rebroadcast message. It could achieve best quality of performance compared to all other existing VANET protocols for safety alert message dissemination. With adaptive broadcast mechanism, the message dissemination flow still has redundant problem.

2.1.2 Clustering/Group

In VANETs, [13]-[17] proposed clusters/groups to provide basis constructions for V2V communications reliability. Several research point out aggregating clusters/groups to increase the anonymity of V2I communications. The clusters/groups communications are automatically determined by vehicle's positions with overlapping groups. Figure 4 displays the group design philosophy [18]. The group leader is located in the center and the geographic group is predetermined with group boundary. However, the formation is not tally with the features of VANETs.

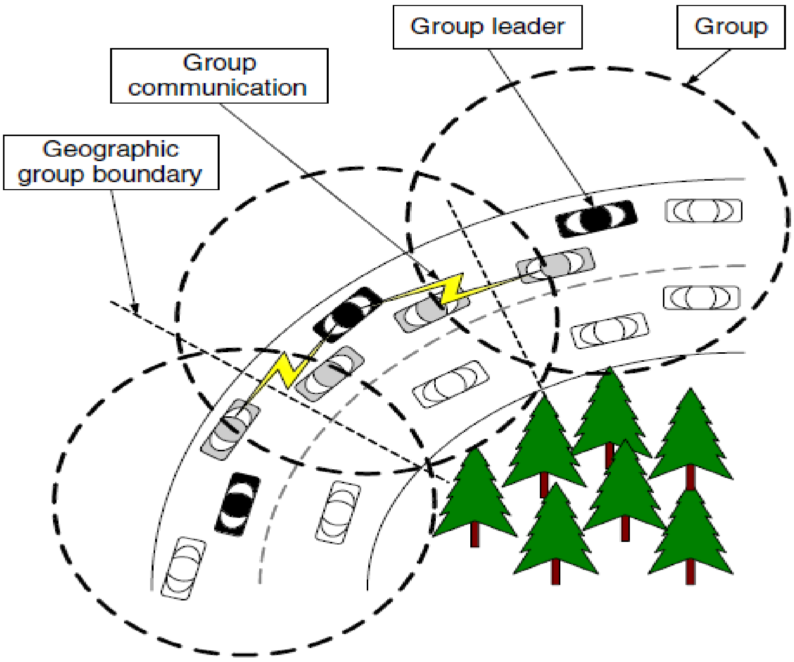

Figure 4: Group formation with geographic boundary [18]

A finite state machine (FSM) is employed to describe how the clusters are operated by state transitions. In [19], the authors present their clustering protocol in Figure 5. There are four states to explain each vehicle enables the proposed clustering protocol and each cluster is organized with three roles: cluster-head, quasi-cluster-head and cluster member.

Inter-cluster control (ICC), Inter-Cluster Data (ICD), cluster range control (CRC), and cluster range data (CRD) are the DSRC channels for the vehicle communications. In addition, each vehicle is equipped with two set of transceivers, T1 and T2.

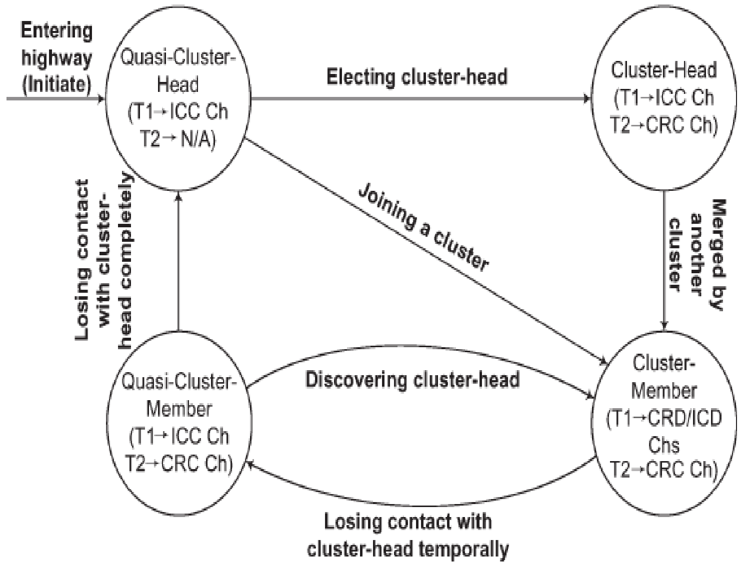

Figure 5: FSM example I [19]

Figure 6 shows the proposed FSM in[20], each cluster is organized with three roles: cluster-head(CH), cluster-tail(CT) and cluster members(CM); there are five conditions for state transitions: initial state, CH contention, CT designation, join contention and losing contact with CH.

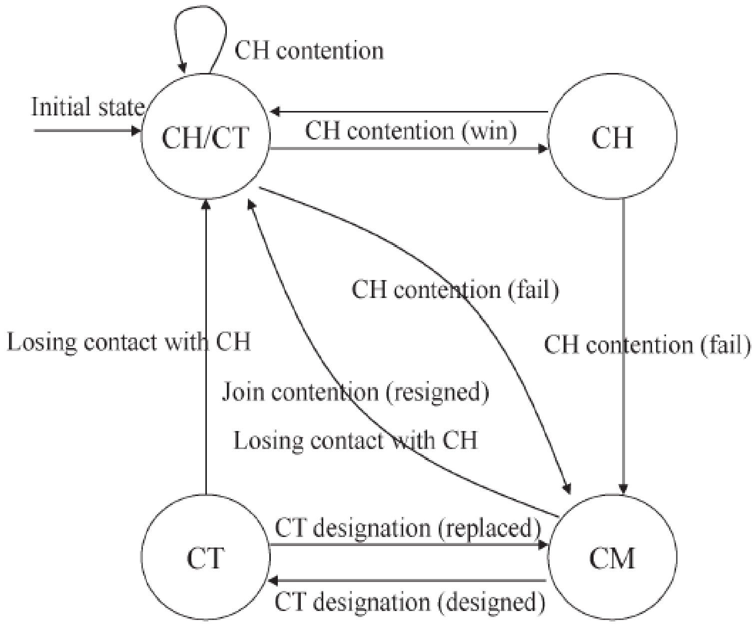

Figure 6: FSM example II [20]

2.2 Distributed Computing Algorithms

In a distributed system, leader election is a very important issue. A leader is responsible of making sure the synchronization, consistency, sequencing and load balance between peers. Many algorithms have been presented for electing leader in distributed systems on networks, such as AUDITOR [21], Bully [22] and Ring [23].

In AUDITOR, each node contains an "auditor" and there is an ordered ranking of auditors. The highest-ranking auditor selects "audit coordinator" which is responsible for detection of failures. The drawback of AUDITOR algorithm is that when multiple nodes are crashed, it may take many sequential executions of the promotion protocol before a candidate is successful in reaching coordinator rank.

In Bully, when a node (e.g. node 6 in Figure 7(a)) detects that the coordinator has crashed and sends an ELECTION message to all nodes with higher numbers. Figure 7(b),(c) and (d) show the message passing actions between each node with higher ID. Figure 7(e) display that the node with highest ID will be the new coordinator.

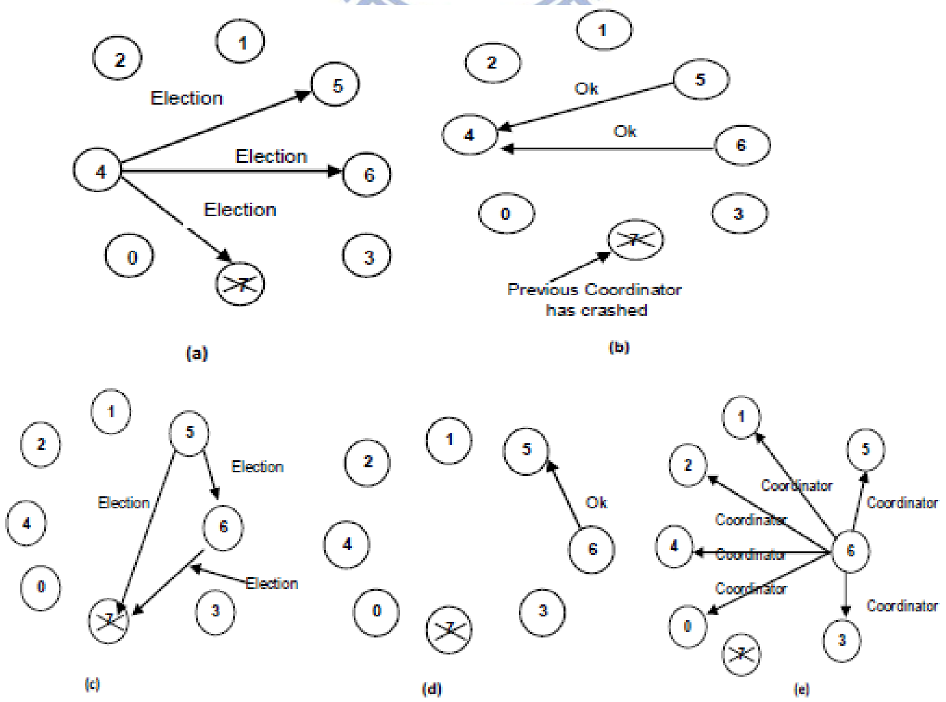

Figure 7: Original Bully Algorithm

The advantage of Bully algorithm is its simplicity, but the main drawback is the high number of message passing. The Big O of Bully algorithm is $O(n^2)$ that increases heavy traffic on the network. Many researches proposed modified Bully algorithms to reduce the number of message passing without discussing the priority definition and consensus in their election process [24],[25],[26].

Paxos[27],[28] is a consensus algorithm for implementing a fault-tolerant distributed system. It is one of the most efficient practical algorithms for achieving consensus in a message-passing algorithm with failure detection mechanisms that allow nodes to exchange messages and maintain information consistency. Paxos describes the actions of the processes by their roles in the protocol: client, acceptor, proposer, learner, and leader. In[29], the leader election algorithm is designed by modifying Bully and Paxos to reach consensus and reduce the number of message passing.

2.3 Real-Time Traffic Information Based on Cellular Network

Real-time traffic information, such as average vehicle speed, travel time, traffic flow, traffic accidents, and other information, can then be referenced by road users and the ministry of transportation to improve the level of service for roadway. At present, the approaches of collecting real-time traffic information can be categorized into three groups:

(1) *Stationary Vehicle Detector* (VD).

(2) *Global Position System* (GPS)-based probe cars reporting.

(3) *Cellular Floating Vehicle Data* (CFVD).

Traffic information has traditionally been gathered by public agencies (Departments of Transportation) via stationary VDs installed in the roadways to detect the average vehicle speed and traffic flow. However, it is quite costly to install and maintain such devices. In addition, the VDs are easily inflicted by temperature fluctuation, moist, and other factors. So they need to be maintained on a seasonal or annually basis.

Alternatively, traffic information can be collected from travelling vehicles equipped with GPS receivers and wireless communication capability as probes on the road network. The GPS-equipped probe cars transmit their positions and speeds to a traffic information center periodically. However, the penetration rate of GPS-based probe cars needs to be high enough to infer more accurate real-time traffic information. In addition, there is extra transmission cost incurred when probe cars send back data through the air periodically. Therefore, the emerging technology, CFVD, which collects and detects the real-time traffic information by tracking the location of *Mobile Stations* (MSs) through cellular network signaling (e.g. *handover* (HO) and *Call Arrival* (CA)) becomes more and more popular for ITS. For example, ITIS Holdings applied the patented CFVD technology for measuring and forecasting real-time traffic information based on anonymously sampling the positions of MSs [30]. As the number of people owning cell phone has increased, it would feasible to use MS as a probe for obtaining traffic information.

2.4 Massive Data Processing Based on Cloud Computing

Cloud computing supports virtualization, on-demand services, scalable flexibility, hardware and software scalability, automatic adaptation, pay-per-use, and service level agreements, among other features[31],[32]. Cloud computing, which is comprised of Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS)[33]. Many computer scientists are predicting that Internet models will move toward a mix of cloud and client-based approaches to meet the growing demand for data-intensive applications [34]. The MapReduce model is an example for data-intensive computing.

2.4.1 MapReduce

MapReduce [35],[36] is an emerging programming model for large-scale data-parallel applications such as web indexing and data mining. The Map functions and Reduce functions are both defined with key/value pairs. The Map functions are distributed across multiple machines by partitioning the input data into a set of M splits automatically. The Reduce invocations are distributed by partitioning the intermediate key space into R pieces using a partitioning function (e.g., hash (key) mod R). Figure 8 shows the overall flow of a MapReduce operation [35].

(1) The MapReduce library in user program shards the input files into M pieces and starts up many copies of the program on a cluster of machines.

(2) The master assigns each worker one a map task or a reduce task. There are M map tasks and R reduce tasks to assign.

(3) The Map function takes an input pair and produces a set of intermediate key/value pairs.

(4) The buffered pairs are written to local disk periodically. In addition, the master is responsible for forwarding the locations of buffered pairs on the local disk to the reduce workers.

(5) The reduce worker read the buffered data from the local disks of the map workers by remote procedure calls. When a reduce worker has read all intermediate data and sorted by the intermediate keys so that all occurrences of the same key are grouped together.

(6) The reduce worker iterates over the sorted intermediate data and for each unique intermediate key encountered. The key and the corresponding set of intermediate values are passed to the user's Reduce function.

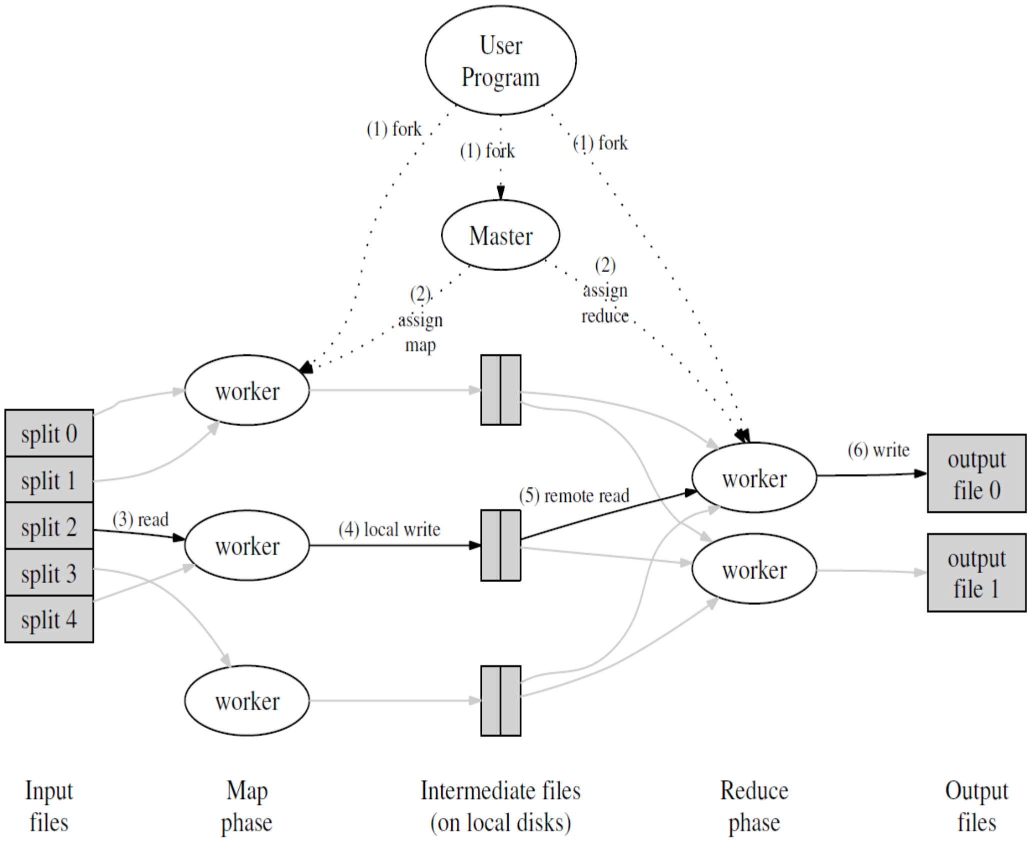

Figure 8: Mapreduce Execution overview [35]

When all map tasks and reduce tasks have been completed, the master wakes up the user program. The output of the MapReduce execution is available in the R output files.

For massive data processing, Google File System (GFS) [37], is a scalable distributed file system for large distributed data-intensive applications. Hadoop is an open-source software project which implements a distributed data processing scheduling and execution environment and framework for MapReduce jobs. Hadoop includes a distributed file system called Hadoop Distributed File System (HDFS)[38] which is analogous to GFS in the Google MapReduce implementation.

2.4.2 Column-Based Data Model (CDM)

Column-Based Data Model (CDM) is a data query model operating on petabytes-level data and can response in seconds. The ideas behind CDM are columnar data layout so query tasks can be executed in place to reduce the execution time. CDM is often used to analyze the outputs of Mapreduce pipelines and prototype larger computations [39]. To maintain the sequences of the data, CDM keeps each NULL in its nested data format. Figure 9 displays the storage format of CDM, it will waste a lot of spaces to record NULLs potentially. To solve this problem, we propose a modified CDM to mitigate it and provide a numeric analysis.

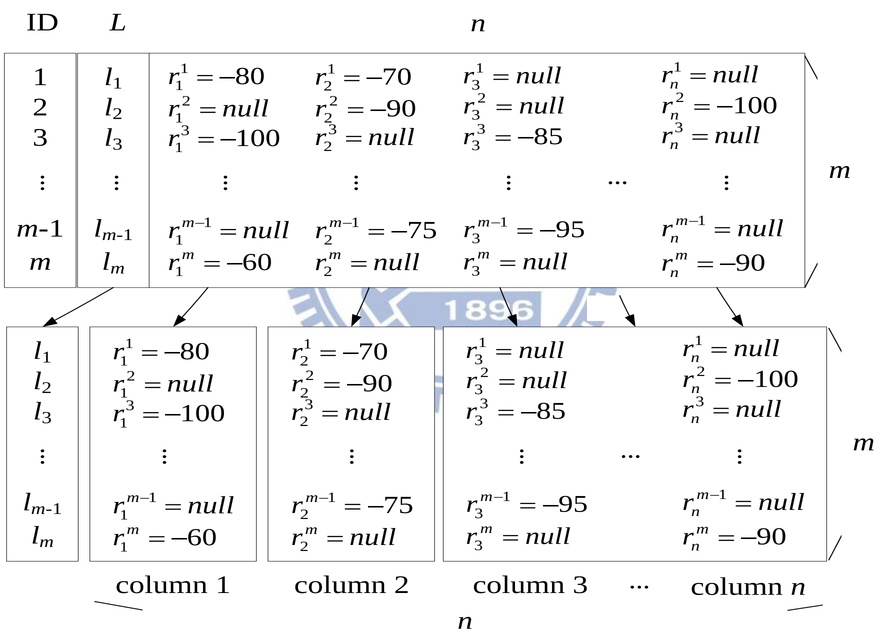

Figure 9: The storage format of CDM

2.4.3 Modified Column-Based Data Model

The modified CDM [40], gives consideration to the storage capacity and search efficiency problems. The storage format of modified CDM is adding ID entity and divided into each column file without the null value as shown in Figure 10.

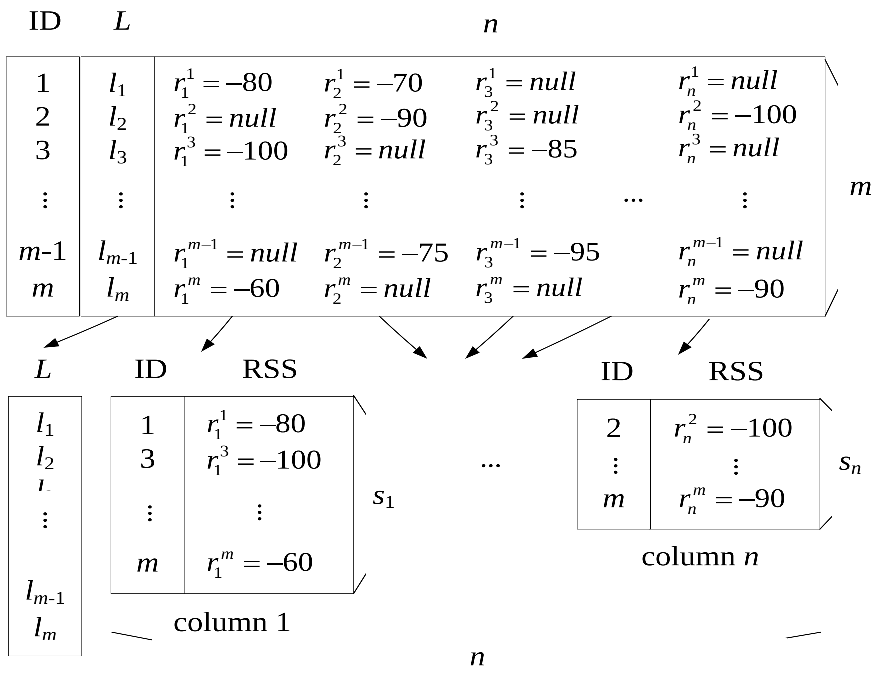

Figure 10: The storage format of the modified CDM

The complexity analysis of CDM and modified CDM are discussed as follows.

- Space Complexity: The space complexity of CDM is $O(nm)$. The modified CDM only stores the non-null value and the ID entity so the space in j-th column file takes $2 \times s_j$ shown as Eq. (2.1). The space complexity of the modified CDM is $O\left(2 \times \sum_{j=1}^{n} s_j\right)$ which is very small and less than the space complexity of CDM in sparse matrices.

$$s_j = \sum_{i=1}^{m} c_j^i, \text{ where } c_j^i = \begin{cases} 1, c_j^i \neq null \\ 0, \text{otherwise} \end{cases} \tag{2.1}$$

- Time Complexity: Searching a non-null value in the j-th column file, the computation time of CDM and modified CDM are expressed as $O(m)$ and $O(s_j)$, where $s_j \leq m$. For reassembling the row values in the other column files, CDM can get these values by using the index of the column file, but the modified CDM can use binary search to get these values with the computation time $O(g)$ shown as Eq. (2.2).

Therefore, the time complexities of CDM and modified CDM are $O(m+1)$ and $O(s_j+g)$, respectively. The time complexity $O(s_j+g)$ is less than $O(m+1)$ in sparse matrices.

$$g = \arg\max_{1 \le b \le n} \log_2\left(s_b\right) \qquad (2.2)$$

In this paper, we will adopt cloud computing technology to improve the performance of massive data processing. The execution of the modified CDM with MapReduce program is to solve the space-wasting and time-wasting problems of massive data processing.

Chapter 3 V2V Communications Model

There are three modes of operation in our traffic information transmission architecture. In this Chapter, we will present the first mode: V2V communications model. The design issues are released in Section 3.1. The symbols and messages are defined in Section 3.2. The consensus-based cluster-head election algorithm will be illustrated in Section 3.3. Finally, the V2V message dissemination model will be presented in Section 3.4.

3.1 Design Issues

Vehicular clusters/groups provide basis for V2V communications in VANETs. For construction a reliable and practical V2V communication, it is vital to transmit and maintain information with consistency. Several research point out that clustering is an efficient technique to reduce the data congestion and keep message effective in dynamical environments. Therefore, we will depict a V2V transmission model based on clusters in a V2V communications environment.

3.2 Notations

Table 1 shows the symbols for message dissemination model.

Table 1: Symbols

Symbol	Descriptions
$ND(i, j, t)$	The ND(i,j,t) is the connection of vehicle j are directly connected to vehicle *i* in time *t*. If there is connection the ND(i,j,t) is 1.
$\sigma_i(t)$	It is the maximum number of vehicles that are directly connected to vehicle *i* in time *t*.
$ARS(i, j)$	The $ARS(i, j)$ is the RSSI between vehicle *i* and vehicle *j*.
$\rho_i(t)$	It is the average RSSI value with other connected node in time t.

N_{ij}	The N_{ij} is the number of the packet signal of vehicle *i* received from vehicle *j*
$Vote_n$	The $Vote_n$ is the Vote of node n. It is used to elect the leader ranking from high to low/
Msg	The total number of the safety alert message dissemination.
Δt_{m2h}	The message transmission time from CM to CH.
Δt_{t2c}	The message transmission time between CT to the hind cluster.
ΔB	The total maximum message broadcast time from the CH to all CMs.
$D(t)$	The total delivery time is from the first message sent out to last cluster received.

Table 2 shows the message for cluster-head election algorithm.

Table 2: Message Types

Message	Descriptions
RTJ	Request to join an existing cluster
CLS_INFO	Cluster information. the CT node is also defined in here
CEHCK_EXIST	CM will send the message to check the existence of CH
ELECTION	The QCM think the CH is exited, QCM will send out ELECTION to start leader election
OK	The node who receive ELECTION will reply OK with Vote
GRANT	The message will send to the node with highest Vote
PROPOSAL	The new CH will use this message to notify CMs the new CH candidate is elected.
ACCEPT	The CM receive PROPOSAL will reply ACCEPT to it.
COORDINATOR	When new CH candidate receive more than half, it will send out COORDINATOR to CMs
ELECTION_STOP	When CM find the CH is still exist, it will send ELECTION_STOP to cancel the leader election process

3.3 Cluster Formation

The process of choosing a cluster-head from the cluster of all members is known as a cluster-head election. We proposed a consensus-based cluster-head election algorithm to elect a leader as cluster-head to maintain and disseminate real-time traffic information to other member vehicles and clusters, which will be introduced in Section 3.3.2

3.3.1 The Finite State Machine

In V2V communication, the proposed finite-state machine (FSM) expresses the vehicle states and describes the operating process of our proposed scheme as shown in Figure 11. Each vehicle operates under one and only one of the following five states at any given time:

(1)Standalone Node (SN),

(2)Cluster-Head (CH),

(3)Cluster-Member (CM),

(4)Cluster-Tail (CT), and

(5)Quasi-Cluster-Member (QCM).

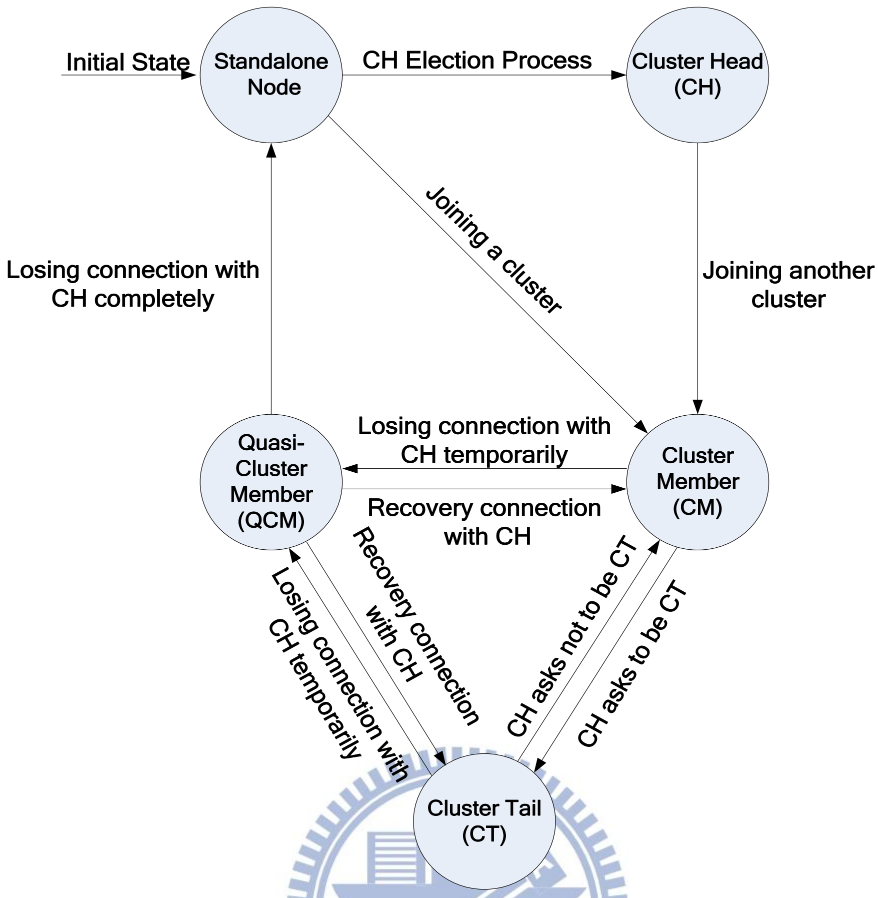

Figure 11: The proposed FSM

The functions of the five states are described as follows.

(1) Each vehicle does not belong to any cluster and prepare to join a cluster called Standalone Node (SN). The SN will send *CEHCK_EXIST* message to find if there has an existing cluster with Cluster Head (CH). The CH and the CMs which received the *CHECK_EXIST* message will reply *CLS_INFO* message to the SN.

(2) When a SN received the respondent *CLS_INFO* message, SN will send *RTJ* (Request to Join) message to join the cluster. The *RTJ* message includes a value called *Vote*. The *Vote* is defined in Section 3.1.2 for cluster-head election. While the SN joins the cluster successfully, it would enter the CM state.

(3) If SN does not receive *CLS_INFO* message, the SN will send out *ELECTION* message to start a cluster-head election process. The SN who is elected to be the new leader will enter CH state. The CH will be responsible for propagating

messages, including the *CLS_INFO* message dissemination in time *t* periodically and the Cluster Tail (CT) assignment of the cluster. Every CM receives the *CLS_INFO* message will reply an *OK* message (Figure 12) that included the updated *Vote* and self-traffic information to the CH. The message delivery of our proposed join process is shown as Figure 13.

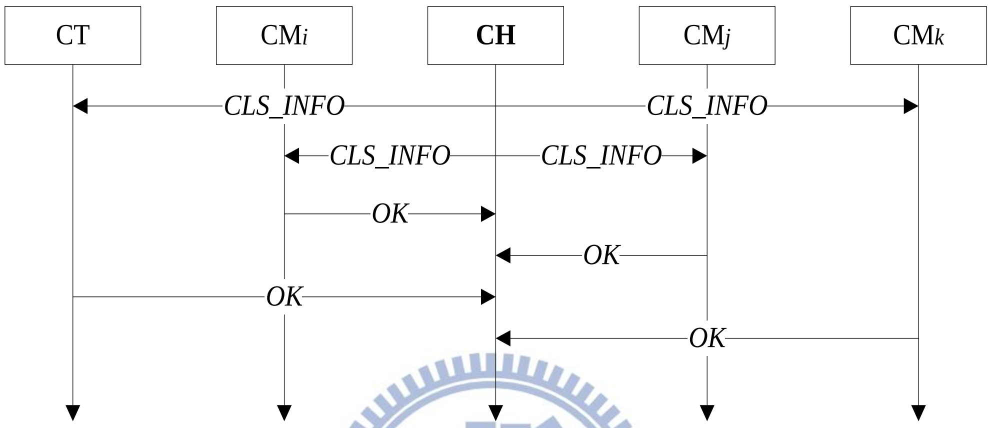

Figure 12: The *CLS_INFO* message broadcast mechanism

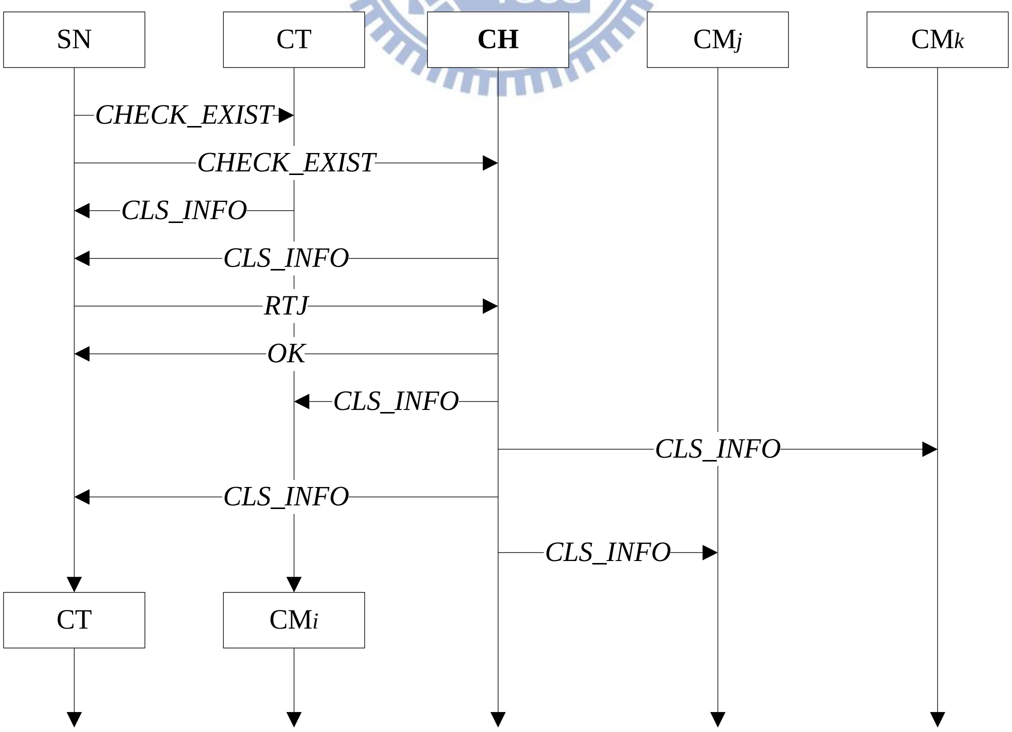

Figure 13: The message delivery of SN joins a new cluster

(4) The CT who is in the rear-end of the cluster and responsible for transferring messages inter-cluster. When the CT receives the broadcast messages from other neighboring clusters, it will send out its keeping messages to them.

(5) When a CM or CT misses the *CLS_INFO* messages over 3 time periods, it will enter the QCM state and send out a *CHECK_EXIST* message to its cluster. There will be waiting for 3 time periods and resending in very period to check the state of the CH. When the CH receives the *CHECK_EXIST* message, it will reply the *CLS_INFO* message to the sender to recover its CM state. If the QCM waits over 3 time periods, it will start a new cluster-head election process to reconstruct the CH role.

3.3.2 Consensus-Based Cluster-Head Election Algorithm

Cluster-head election is an influential problem for clustering in VANETs. A cluster-head plays an important role in coordinating and maintaining the cluster. Therefore, we proposed a consensus-based cluster-head election algorithm which is based on Bully [22] and Paxos [27],[28] algorithms. The proposed algorithm guarantees to reach consensus, fault-tolerant and reduce the number of message passing in the cluster-head election process. The proposed consensus-based cluster-head election algorithm has the following three steps:

1. Calculating the cluster-head election criteria;
2. Cluster-head election algorithm
3. New cluster-head announcing

3.3.2.1 Cluster-Head Election Criteria

In the Bully algorithm, the identity number (ID), will be the weight of each node in the election process which is generated by the system. For this reason, we generate a new ID method by node degree and signal strength as the cluster-head election criteria. We propose node degree and signal strength to compute a specific ID that will be used as *Vote* in the cluster-head election process. The highest voted node will be elected as the CH.

- Node Degree (ND): The overall node degree is the maximum number of vehicles that are directly connected to vehicle *i* in time *t*. It expresses as $\sigma_i(t)$ and defined as Eq. (3.1).

$$\sigma_i(t) = \sum_j ND(i, j, t) \tag{3.1}$$

Where *j* is a potential neighboring vehicle. The $ND(i, j, t)$ is equal to 1 if a connection between *i* and *j* exists at time *t*, and is equal to 0 otherwise.

We use node degree as a benefit criterion in order to make the CH with most connectivity in the cluster. V2V refers to the direct or multi-hop communications among vehicles. So every vehicle has more connected node degree would have less packet loss and better radio coverage. Therefore, the CH will be with the highest node degree to have less maintained cost when it is elected.

- Average RSSI: The Receive Signal Strength Indicator (RSSI)[41] is the value of the received packet signal. The CH with higher average RSSI will have less chance to lose the connection with other CMs. It expresses as $\rho_i(t)$ and defined as Eq.(3.2)

$$\rho_i(t) = \frac{\sum_n \left(\sum_j ARS(i, j) \right)}{N_{ij}} \tag{3.2}$$

Where *j* is the neighboring vehicle that sends package to vehicle *i*. The $ARS(i, j)$ is the RSSI between vehicle *i* and vehicle *j*. The N_{ij} is the number of the packet signal of vehicle *i* received from vehicle *j*.

Every SN received *ELECTION* message would reply the *Vote* in *OK* message to the finder, who is the first detector finding the CH is disconnected. The *Vote* of vehicle n is calculated as Eq.(3.3):

$$Vote_n = \sigma_1(n) + \rho_i(t) \tag{3.3}$$

The $Vote_n$ ranking also represents the priority in the cluster. With the highest $Vote$ in the cluster-head election process, the SN/QCM/CM will be elected as the CH.

3.3.2.2 Cluster-Head Election Algorithm

The proposed consensus-based cluster-head election algorithm is depicted as follows. When a QCM detects the CH has left its cluster or joined to other cluster, it will send out the *ELECTION* messages to the higher priority CMs and start the cluster-head election process as shown in Figure 14. The scenario shows that the CM3 is the finder who detects the CH has left the cluster and sends out *ELECTION* messages to superiors (CM4-CM6) immediately. When every superior CM receives the *ELECTION* message will also send *CHECK_EXIST* message to make sure the existence of the CH (Figure 15).

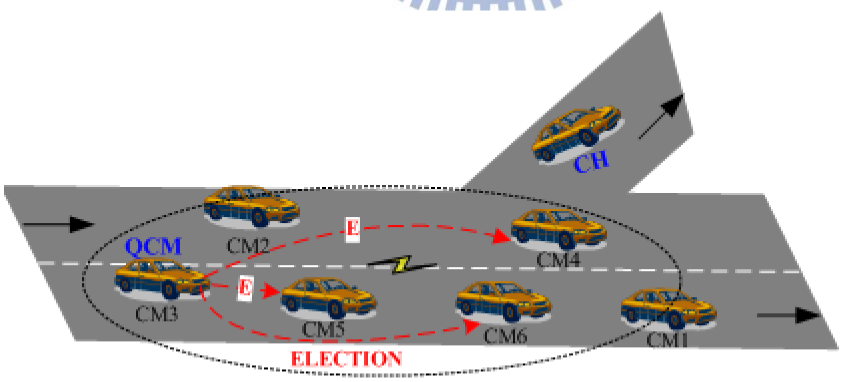

Figure 14: Detecting the CH is disconnect with the cluster

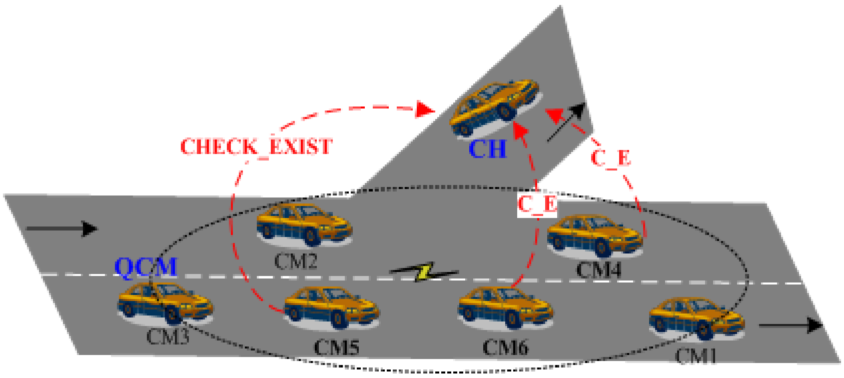

Figure 15: The CHECK_EXIST message

After waiting a period time without any response from the CH, all of received *ELECTION* message CMs will reply *OK* message to the QCM (Figure 16). As receiving the *OK* messages from the superior CMs, the QCM is responsible for sending the *GRANT* message to the superior CM whose priority number is the highest at present (Figure 17). The message delivery process of the cluster-head election algorithm is shown in Figure 18.

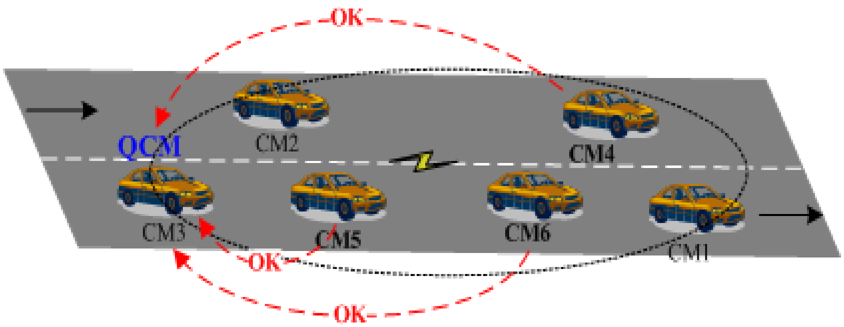

Figure 16: The OK message

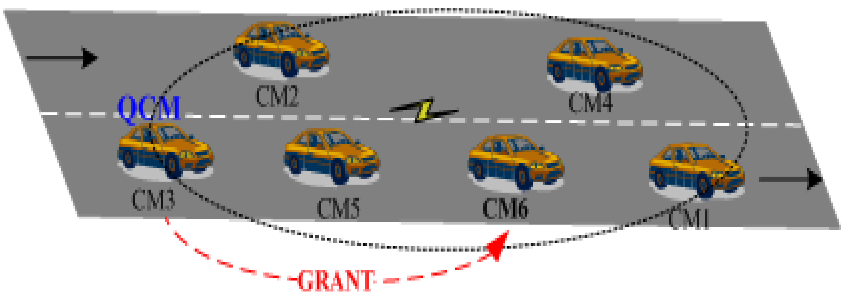

Figure 17: The GRANT message

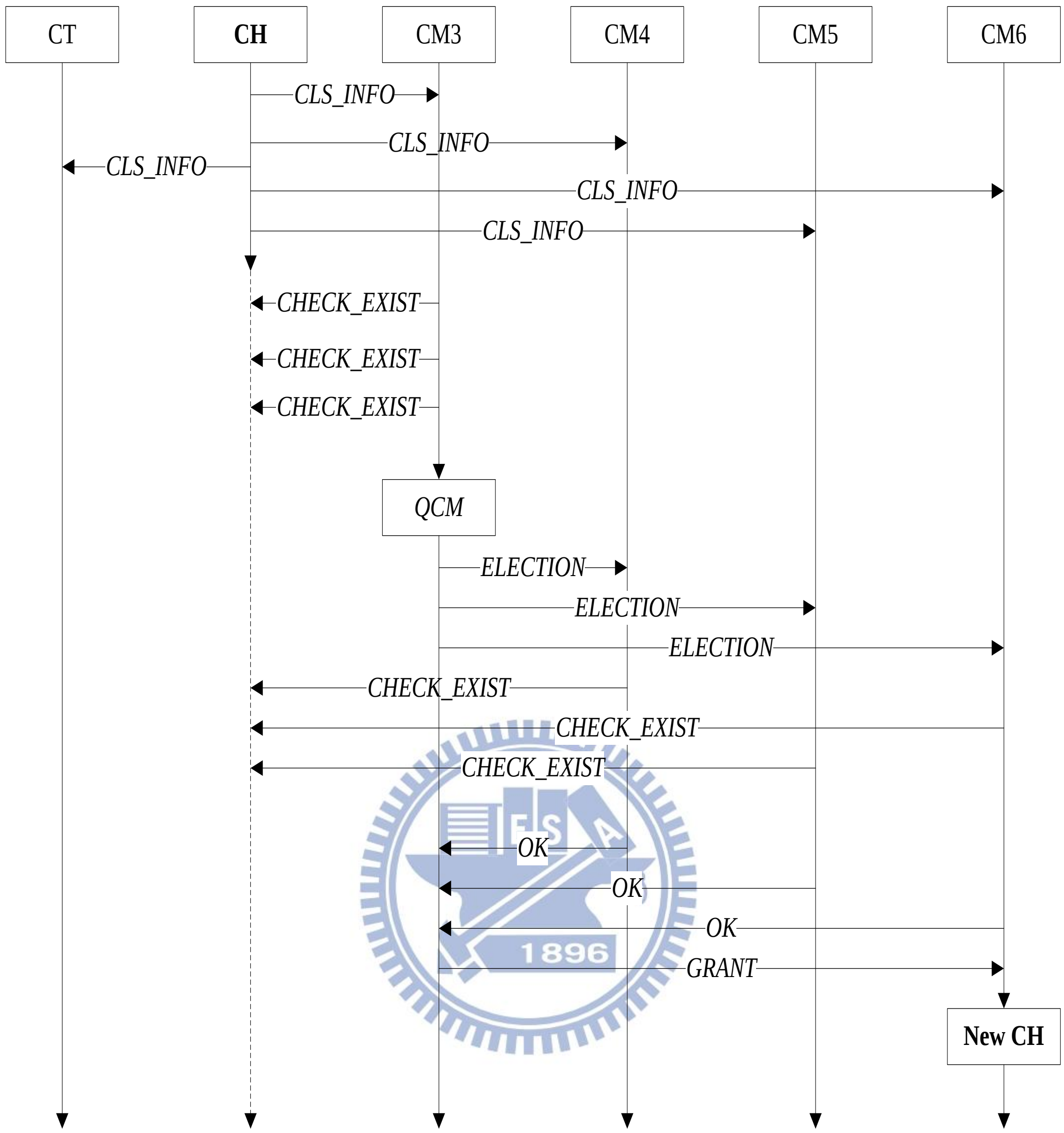

Figure 18: The message delivery of the cluster-head election algorithm

Figure 19 shows the granted CH (CM6) broadcasts the *PROPOSAL* message to remaining CMs in the cluster. Every CM receives the *PROPOSAL* message has to reply *ACCEPT* message to recognize the new CH (Figure 20) When more than half remaining CMs reply the *ACCEPT* messages to the new CH, it will broadcast the *COORDINATOR* messages to declare himself to be the new CH (Figure 21).

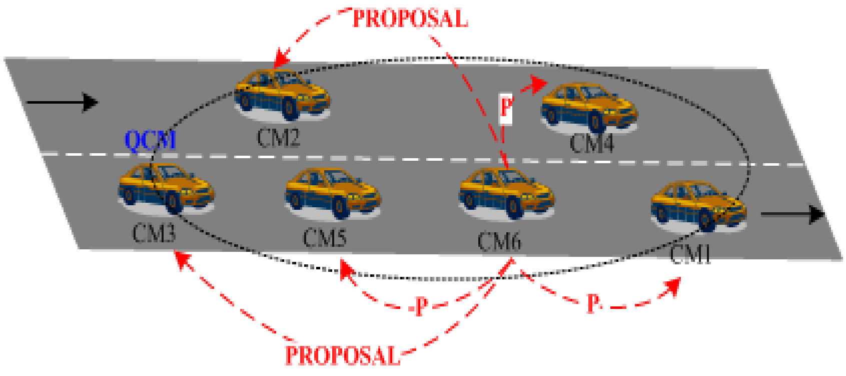

Figure 19: The PROPOSAL message

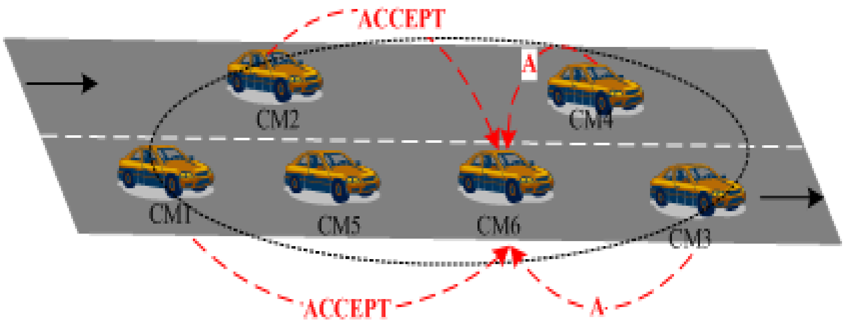

Figure 20: The ACCEPT message

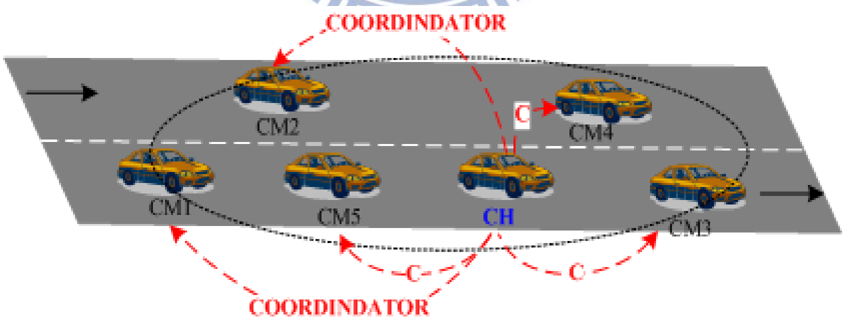

Figure 21: The COORDINATOR message

The pseudo code of the cluster-head election process is presented as Pseudo Code 1.

```
A node finds that cluster-head is crashed
Start the cluster-head election process
New CH_ is null
For each node in the network/system
        Send out ELECTION
Wait for nodes reply
For each reply in the group
{
        Get the highest priority and set node to be New_CH;
}
Send the GRANT to the New_CH
Wait for CH PROPOSAL and COORDINATOR
```

Pseudo Code 1: The cluster-head election process

There are two possible cases may occur during the cluster-head election process. The detail descriptions are as follow.

(1) The CH is still alive when one SN/QCM starts cluster-head election process

When a CM changes its state to the QCM state, or a QCM changes its state to the SN state, it sends the *ELECTION* message to the SNs/CMs with higher priority number. In VANETs, the package lost or radio coverage might affect the connection between the CH and CMs. When a CM/QCM receives the *ELECTION* message, it will send *CHECK_EXIST* message to the CH. If the CH is connecting, the *OK* message will be sent to the sender. When the node receives *OK* message from CH, it will send an *ELECTION_STOP* message to the QCM which starts group reelection protocol. Once the node gets an *ELECTION_STOP* message, it will recover from the CM state.

If the CH is indeed left, all the nodes received the *ELETCION* messages will reply OK messages to the finder node. The CM replied *OK* message will change to be

SN and send out *ELECTION* message to the QCM. The QCM will wait for 3 average times to grant the highest priority one to start the CH announcing protocol. Because the node received the *ELECTION* message will send out a *CHECK_EXIST* message and wait for an average time. The QCM wait for 3 average times is more reliable way to send the *GRANT* message. The SN had replied the *OK* message to the QCM and will wait for a *GRANT* message. When it receives the *GRANT* message, it will start a consensus CH announcing protocol. In VANETs, the consensus will be achieved and reduce many maintain cost in this unreliable environment.

(2) More than one CM detects the CH has left

When more than one CM changes its state to QCMs, they will send the *ELECTION* message to the node with higher priority. If a QCM sends out *ELECTION* message and receive the *ELECTION* message from other nodes at the same time. The higher priority QCM will send *ELECTION_STOP* message to ask the lower priority one to stop cluster-head election process. The node received the *ELECTION_STOP* message will stop the cluster-head election protocol immediately.

3.3.2.3 CH Announcing Process

When a SN received the *GRANT* message will be the nwe CH. The CH is going to be the proposer. The proposer will send its new proposal and announce that he is going to be the CH in this cluster. All the rest SNs will be an acceptor and wait for the *PROPOSAL* message. The acceptors agree the proposal and reply to proposer. When more than half nodes in the group reply to proposer, the proposer will make announce to all the nodes to be the CH. After the announce sending, the cluster-head election protocol is done. The message delivery of the new leader announcing is shown in Figure 22.

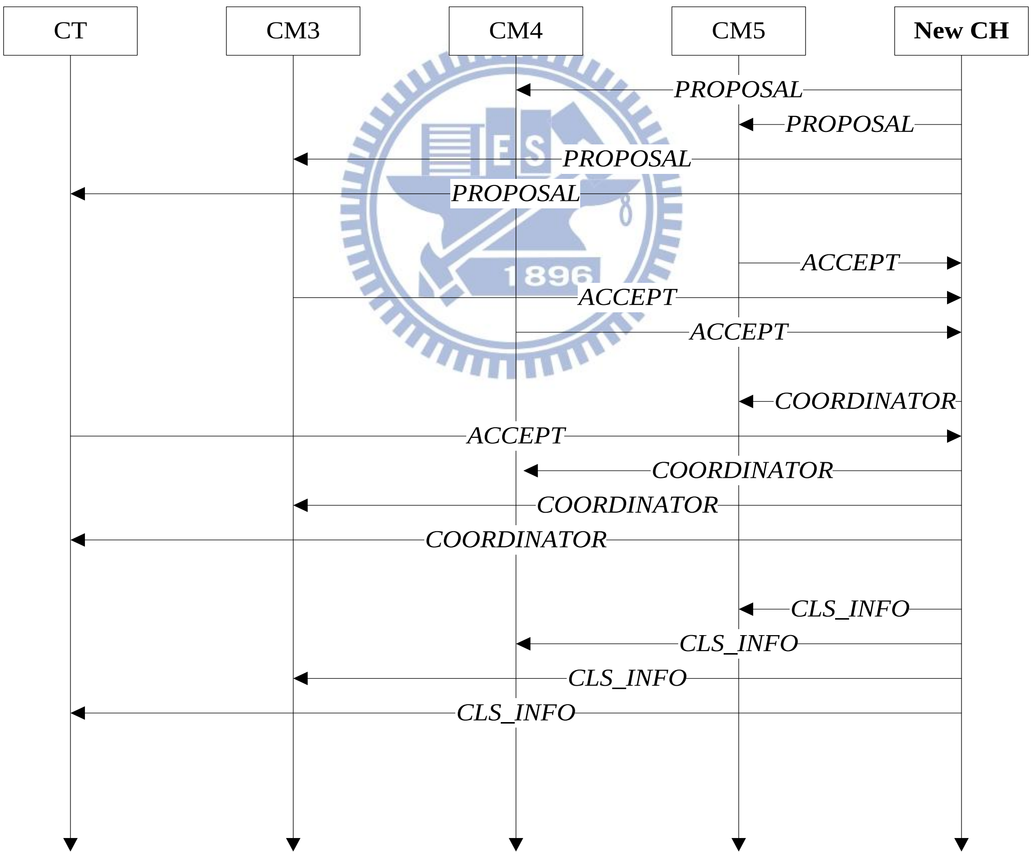

Figure 22: The message delivery of the new leader announcing

When more than one node send out the *PROPOSAL* messages at the same process time, each acceptor will accept the *PROPOSAL* message with higher priority number than itself. For example, one acceptor receives a *PROPOSAL* message with priority 6. It will reply an *ACCEPT* message to the proposer. In the meantime, it receives another *PROPOSAL* message with priority 5. The acceptor will reply a message to sender with priority 5 and say: " There is a SN to be the CH. His priority is higher than you, please stop sending the message." Moreover, the acceptor receives the other *PROPOSAL* message with higher priority 8, it will accept the *PROPOSAL* message and send a message with priority 7 to sender to stop the CH announcing.

The Pesudo Code 2 shows the new CH announcing process:

```
Receive PROPOSAL:
If( Current_Proposal is null)
        Reply APPROVE message and set Current_Proposal as sender;
Else
{
        If(Current_Proposal is higher than sender)
            Reply DENY message to sender with the Current_Proposal;
        Else
        {
              Reply APPROVE message to sender;
              Reply DENY message to Current_Proposal;
              Set Current_Proposal to be sender;
        }
}
Receive ACCEPT:
Increase TOTOAL_ACCEPT_NUMBER;
If(TOTOAL_ACCEPT_NUMBER > (TOTAL_MEMBER_NUMBER/2))
{
        For each node in the network/system;
        Send the COORINDATOR message;
}
Else
        Wait for other reply;
```

Pseudo Code 2: The new CH announcing process

3.4 V2V Message Dissemination

There are two main working roles, the CH and the CT, in our proposed V2V message dissemination model. The CH is responsible for broadcasting messages to CMs and choosing a CT at the tail of the cluster. The information of which the node is chosen to be CT is included in *CLS_INFO* message. When the CT receives a message from the CH, it is responsible transferring messages to the back cluster if it receives a broadcasting message from the front cluster. The real-time traffic information, which includes traffic speed, traffic flow, traffic density, traffic accidents/incidents, and other information, can be referenced by road users. In this V2V communications mode, the real-time traffic data is focus on the traffic accident information dissemination. When a car accident event had occurred, an alert message can be warning out from that accident car and transfer to the CH as shown in Figure 23a)(b).

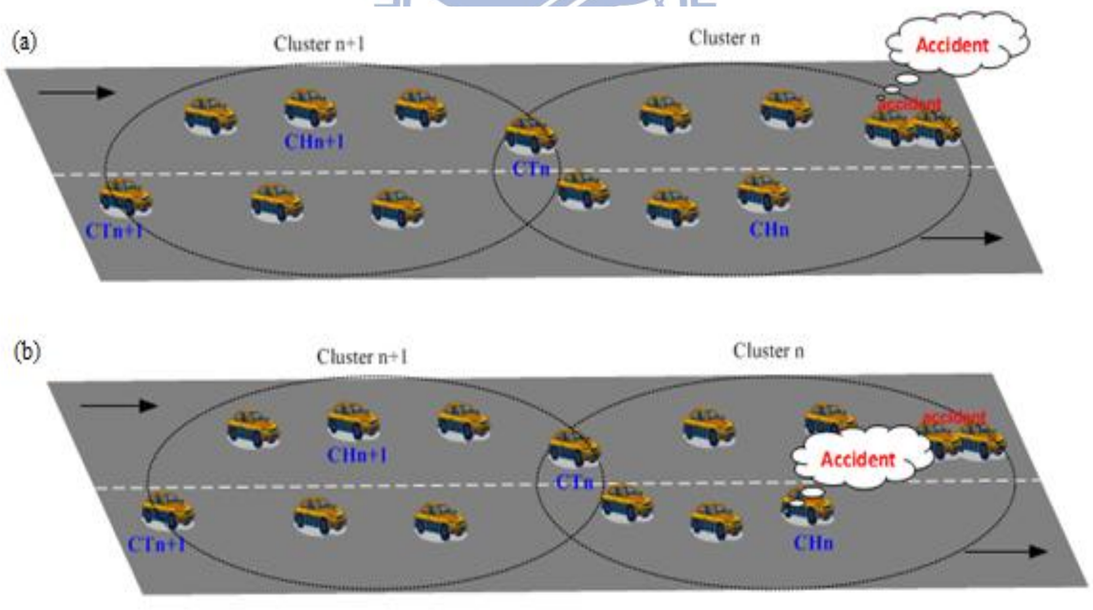

Figure 23:(a) A safety alert message is warming out and; (b) The alert message has transmitted to the CH.

The CH will broadcast the safety alter message to CMs to help the on-coming vehicles avoid the congestion. The proposed safety alert message dissemination model is to propagate adequate and effective message inter-cluster. In our proposed message dissemination model, the CH and CT are responsible for the message dissemination inter the cluster. For inter-cluster communication, the Figure 24a)(b) show that the CT transfers message to the neighboring cluster. In our design, the CT is predefined by the CH and noticed by broadcasting in *CLS_INFO* message. The proposed safety alert message dissemination process in V2V mode improves the efficiency of inter-cluster communication.

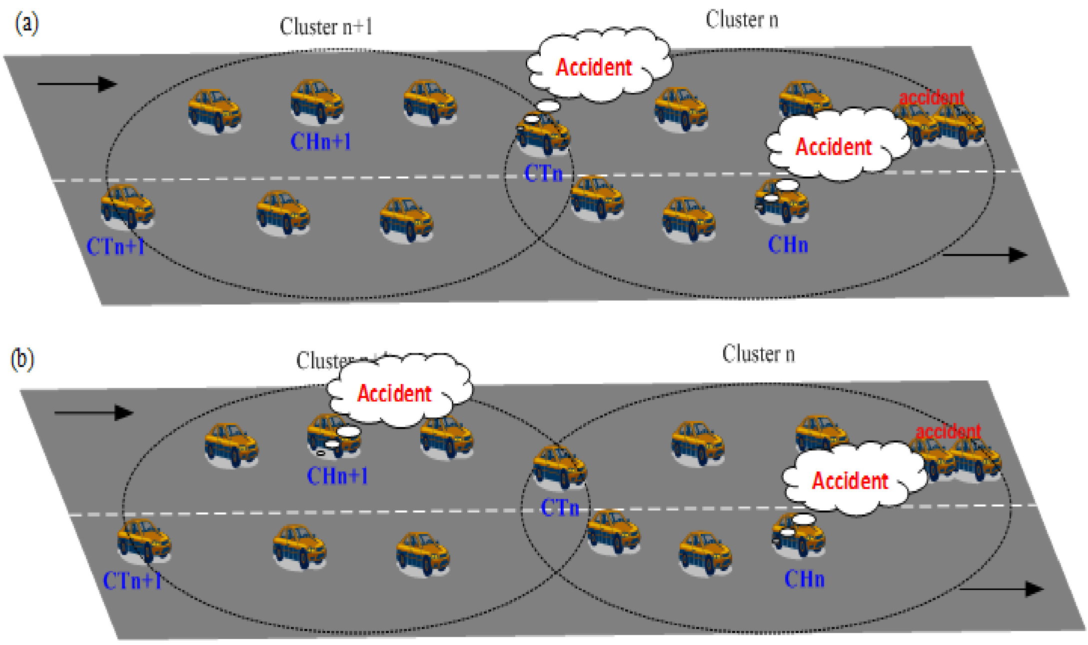

Figure 24:(a)The CH sends the message to the CT in cluster n; (b)The inter-cluster communication via CT

Figure 25 and Pesudo Code 3 display the message dissemination process and pseudo code in V2V communications mode respectively.

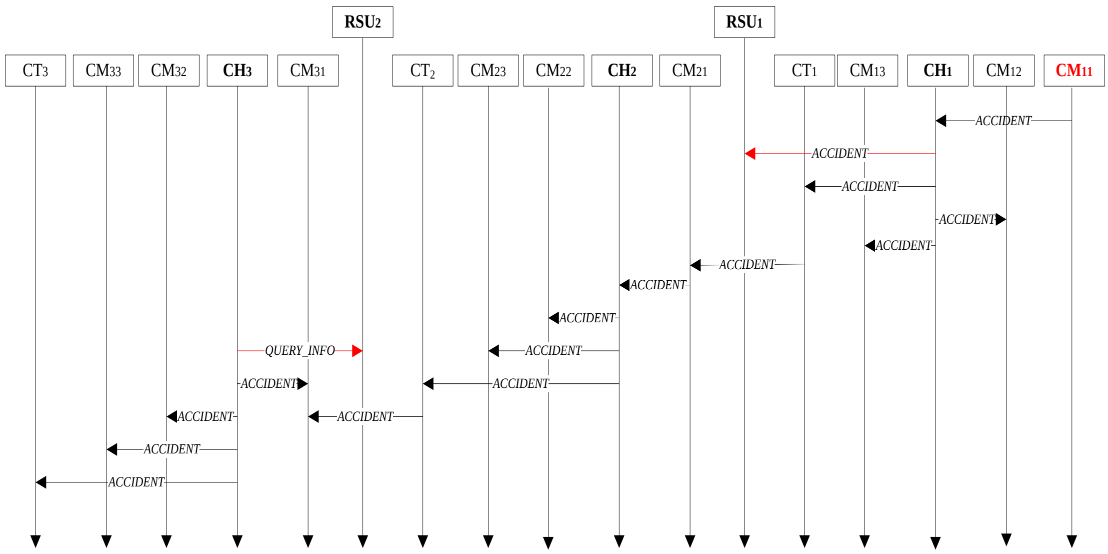

Figure 25: The message dissemination process in V2V communications mode

```
Receive Message:
Switch(Role)
{
Case CH:
        Send Message to CM;
        Break;
Case CT:
        If(Message.own == CH)
                Receive message;
        Else
                Transfer received message to the hind cluster;
        Break;
Case CM:
        If(Message.own ==CH)
                Receive message;
        Else
                Transfer to CH;
        Break;
Default:
        Break;
}
```

Pseudo Code 3: The message dissemination in V2V communications mode

We will use mathematic analysis to calculate the number of message dissemination and the message deliver time in this proposed mode. There are J clusters to transmit and receive message in the same driving direction. In addition, each vehicle is within the transmission rage of its forwarder. When an accident occurred in the first cluster, Msg is expressed as the total number of this safety alert message dissemination from accident spot to the J_{th} cluster. The CTs take $(J-1)$ times to transfer message to its hind cluster.

In a well-connected environment, we assume that every CT could disseminate message to the CH of its hind cluster. The total number of the safety message dissemination in well-connected environment could be expressed as the following:

$$Msg = 1 + J + (J-1) = 2J$$

In a discrete environment, CT will broadcast the message to the hind cluster. There is a probability of the next cluster member received the message from front cluster, P_n. The number of the cluster member could be express as M_n. The CM who receives message from CT could be expressed as $P_n * M_n$. The total number of the safety message dissemination in the environment could be expressed as the following:

$$Msg = 1 + J + (J-1) + \sum_{0}^{j-1}(P_n * M_n) = 2J + \sum_{0}^{j-1}(P_n * M_n)$$

The message transmission time from CM to CH is written as Δt_{m2h}. The total maximum message broadcast time from the CH to all CMs is defined as ΔB. The message transmission time between CT to the hind cluster is written as Δt_{t2c}. The delivery time $D(t)$ would be included: the first message sends out from the accident vehicle to the CH, the CH broadcasts to all CMs, $(J-1)$ times of CT sends to the hind cluster. In a well-connected environment, the D(t) is as below.

$$D(t) = \Delta t_{m2h} + \Delta B + (J-1)(\Delta t_{t2c}) + (J-1)\Delta B$$

$$D(t) = \Delta t_{m2h} + (J-1)(\Delta t_{t2c}) + (J)\Delta B$$

When in a discrete environment, the message deliver time is as below:

$$D(t) = \Delta t_{m2h} + \Delta B + (J-1)(\Delta t_{t2c}) + (J-1)(\Delta B + \Delta t_{m2h})$$

$$D(t) = J(\Delta t_{m2h}) + J(\Delta B) + (J-1)(\Delta t_{t2c})$$

Chapter 4 V2I Communications Model

V2I communications model is designed to enable the transmission of traffic information over a wireless network which enables roadside units (RSUs) to perform calculations and issue drivers advisories to mitigate traffic congestion or avoid crashes through specific safety applications. In this section, we consider the approach of collecting real-time traffic information by tracking the locations of each MS through the cellular networks. The architecture is presented in Figure 26[42]. The BSC of GSM/GPRS and the RNC of UMTS will receive measurement reports from active cell phones during the calls. In Section 4.1, we will analyze the measurement reports and adopt the fingerprint positioning algorithm (FPA) to calculate the average speed of the moving cell phones in a specific road segment. Section 4.2 will illustrate the speed estimation. Moreover, for the massive data processing issue from using FPA, we adopt the MapReduce and modified CDM to solve the space-wasting and time-wasting problems which are due to the sparse matrices generated by FPA in Section 4.3.

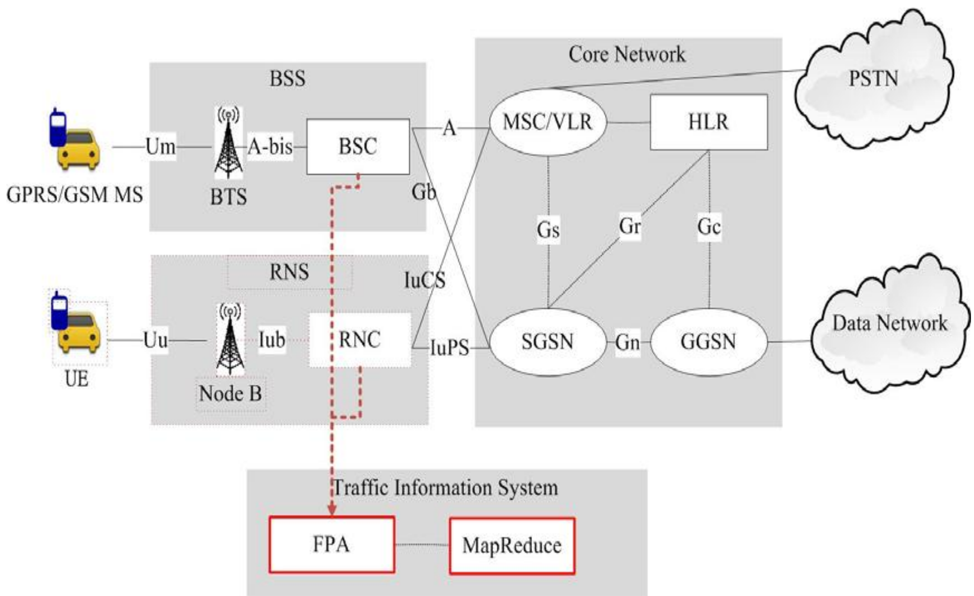

Figure 26: The traffic information system based on cellular network [42]

4.1 Fingerprint Positioning Algorithm

In [40],[42],[43],[44] the kNN-based fingerprint position algorithm (FPA) for traffic speed estimation is proposed. For traffic information estimation, we collect the received signal strengths (RSS) of measurement reports from each MS when the MS is in active mode (e.g., handover (HO) and call arrival (CA)). The RSS are then analyzed by FPA to determine the location of MS, and the MS speed is estimated from more than two locations.

We are given a set of beacons $B = \{b_1, b_2, \ldots, b_n\}$ in a field, which are capable of transmitting radio signals from n base stations (BSs), and a set of training locations $L = \{l_1, l_2, \ldots, l_m\}$. At each training location l_i, we measure the signal strengths from beacons for a period of time and create a characteristic vector $r_i = \{r_1^i, r_2^i, \ldots, r_n^i\}$ in a location database, where r_j^i is derived from the RSS of b_j, $j = 1, \ldots, n$. When an object moves into the field, it also measures its RSS vector $x = \{x_1, x_2, \ldots, x_n\}$ and compares the value against the database to determine its location.

The Euclidean distance between signal vectors is used to determine a location $l(x)$ from each scrambled vector x. For each and each training location l_i's characteristic vector $r_i = \{r_1^i, r_2^i, \ldots, r_n^i\}$, the distance is calculated by Eq. (4.1).

$$d(x, l_i) = \sqrt{\sum_{j=1}^{n} \left(x_j - r_j^i\right)^2} \tag{4.1}$$

The training location with the minimum Euclidean distance will be chosen as the nearest location w_1 by using Eq. (4.2).

$$w_1 = \arg\min_{l_i \in L} d(x, l_i) \tag{4.2}$$

Finally, we will choose the k nearest locations $w = \{w_1, w_2, \ldots, w_n\}$ via the distance-weighted k-nearest-neighbor rule [45]. The predicted location $l(x)$ is calculated by Eq. (4.3).

$$l(x)=\frac{\sum_{a=1}^{k}\left[\frac{d(x,w_k)-d(x,w_a)}{d(x,w_k)-d(x,w_1)}\times w_a\right]}{\sum_{a=1}^{k}\frac{d(x,w_k)-d(x,w_a)}{d(x,w_k)-d(x,w_1)}} \tag{4.3}$$

4.2 Speed Estimation

A novel speed estimation method for the determination of speed information of a MS by means of a RSS characterized in that the calculation is performed by a cellular network executing the steps as follows.

(1) Determining a first location $l(x_1)$ of the MS.

(2) Determining a second location $l(x_2)$ of the MS.

(3) Determining the time difference $[t_1,t_2]$ between the determination of first location $l(x_1)$ and second location $l(x_2)$.

(4) Determining the speed $u_{1,2}$ of the MS based on the time difference $[t_1,t_2]$ and the geographic dictation $D(l(x_1),l(x_2))$ of the first location $l(x_1)$ and the second location $l(x_2)$ using Eq. (4.4).

$$u_{1,2}=\frac{D(l(x_1),l(x_2))}{[t_1,t_2]} \tag{4.4}$$

4.3 Accelerative Processing in Cloud Computing

The FPA position approach generates a big table with locations as the rows and the RSS vectors from neighboring BSs as the columns. The training set can be expressed as a $m\times n$ matrix, is a large number of complex signals from cellular networks. As the number of the training locations and BSs has increased, it will take much cost to process the massive data. As mentioned previously, the modified CDM gives consideration to the storage capacity and search efficiency. For this reason, we implement the MapReduce model and adopt modified CDM to mitigate the space-demanding and time-consuming problems. The complexity comparisons between CDM and modified CDM are as follows.

(1) Space Complexity: There are a large number (m) of training locations with their RSS vectors which are detected from the n different BSs. CDM approach stores all RSS vectors of each BS, and the space complexity is $O(mn)$. However, the modified CDM only stores the non-null RSS vectors and location's ID so it takes $2 \times s_j$ space in j-th column file shown as Eq.(2.1). The space complexity of the modified CDM is $O\left(2 \times \sum_{j=1}^{n} s_j\right)$.

(2) Time Complexity: With the computation time $O(g)$ shown as Eq. (2.2), the time complexities of CDM and the modified CDM are $O(m+1)$ and $O(s_j + g)$, respectively. Therefore, the time complexity of modified CDM is less than CDM in sparse matrices.

Next, we build up a training table which record every location l_i (including the identification and geographic coordinates) and its collection of signal strengths $r_i = \{r_1^i, r_2^i, ..., r_n^i\}$. For a location with a signal set $x = \{x_1, x_2, ..., x_n\}$ to be determined, the Euclidean distance $d(x, l_i)$ is processed in Hadoop. As shown in Figure 27, the executions of mapper and reducer functions include the following:

(1) For design and build up mapper and reducer functions, we set the location l_i as the key, the signal set $r_i = \{r_1^i, r_2^i, ..., r_n^i\}$ and the testing signal set $x = \{x_1, x_2, ..., x_n\}$ as the values.

(2) The training datasets will be partitioned into smaller ones for mappers to process independently. The Mapper calculates the Euclidean distance between r_i and x and generates the intermediate <key, value> pairs <l_i, $d(x, l_i)$> by Eq. (4.1). The temporary results are stored in HDFS and waiting for reducer to further processing.

(3) The Reducer sorts the intermediate <key, value> pairs and selects the smallest k pairs to determine the location $l(x)$ using Eq. (4.3). When there are more than two locations of the same MS, the speed estimation is then computed by Eq. (4.4).

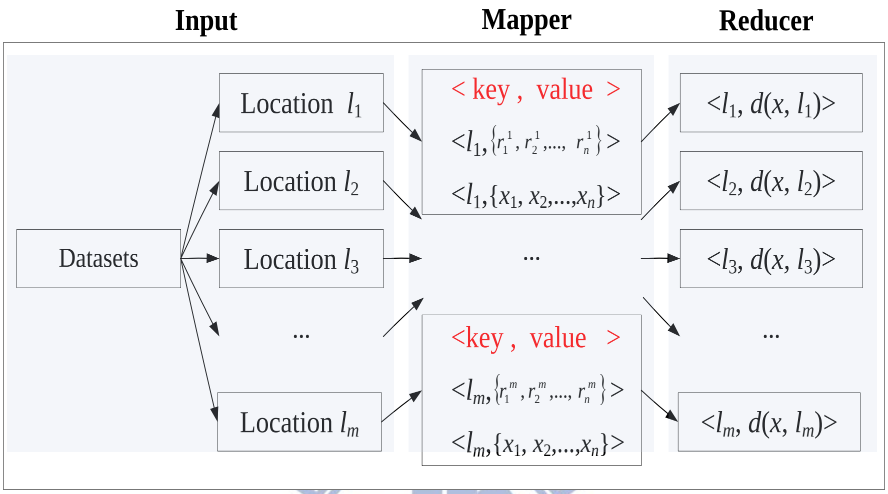

Figure 27: An overview of mapper and reducer function

Therefore, the proposed V2V communication mode takes responsible for estimating safety related information (e.g., traffic speed and traffic flow) from cellular networks and transmitting real-time data between RSUs.

Chapter 5 Hybrid Communications Model

A hybrid communications model for traffic information dissemination in VANETs is introduced in this Chapter. The design issues are released in Section 5.1. In Section 5.2 presents the hybrid communications model and the hybrid message dissemination model will be presented in Section 5.3.

5.1 Design Issues

VANETs are based on short-range wireless communication between vehicles. The allocation of 75 MHz in the 5.9 GHz band is licensed for DSRC which aimed at enhancing bandwidth and reducing latency for V2V and V2I communication. Traffic information that are transmitted over the VANETs can divided into real-time (e.g. safety alert and video signals) and non-real-time (e.g. weather, e-map, and travel information). In this research, we propose a hybrid, V2V+V2I, message dissemination model for transmitting the real-time traffic information in VANETs. It designs to take over the broadcast storm and frequently disconnected transmission problems.

5.2 Hybrid Communications Model:V2V and V2I

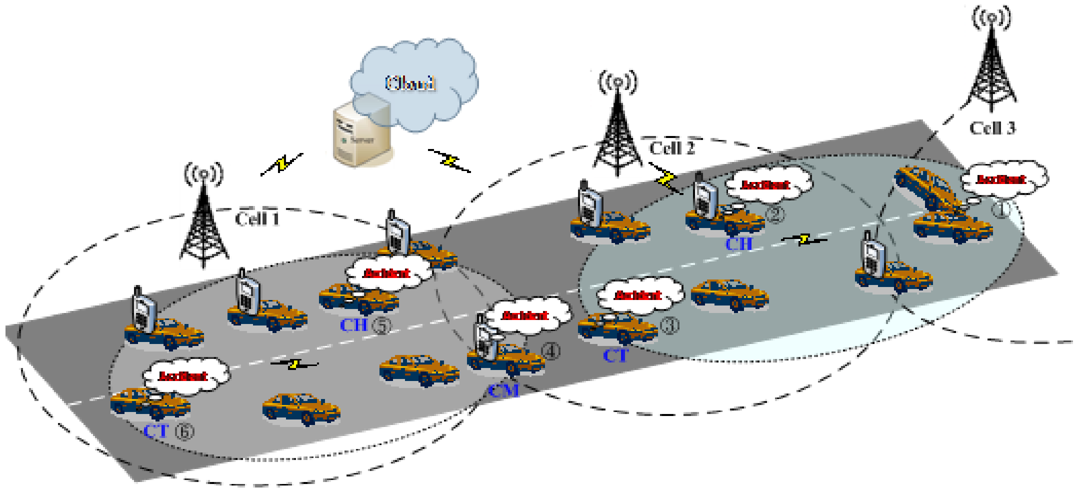

Figure 28: The hybrid communications model

The hybrid communications model, V2V+V2I, is shown as Figure 28. There are six vehicular states to describe the message dissemination in hybrid model. When an accident occurred, the safety alert message is warming and the following as:

State 1: The message dissemination from the accident spot to the CH1 is via V2V model;

State 2: The CH1 broadcast to the CMs and CT1 is responsible for transferring to the hind cluster via V2V model;

State 3: The CT1 is disconnected to the hind cluster; the neighboring clusters are not within the reliable transmission range.

State 4: The CH1 will upload the message to cloud via V2I model (RSU in cell2);

State 5: The CH2 will obtain the latest message on cloud via V2I model (RSU in cell1);

State 6: The CH2 broadcast to the CMs and CT2 is responsible for transferring to the hind cluster via V2V model.

5.3 Hybrid Message Dissemination Model

With our hybrid message dissemination model, we combine V2V and V2I communications models to take over this problem. With roadside unit, CH will receive the message from RSU to help transferring message. The CH is also design to be requested traffic information from RSU. With the RSU help, the message dissemination time will be shortened in larger range.

As the same scenario presented in Section 3.4, the hybrid message dissemination starts up when a car accident occurred. With the RSUs help, the safety alert message would be requested from RSU in some clusters. In our simulation environment, when the message is update to could after 10 sec. every CH will have the message from RSU instead of transferring by other clusters. It shortens the message transferring time between cluster and cluster. Figure 29 resents the scenario diagram for message dissemination from the accident spot (CM_{11}) to hind clusters. CH_1 will upload the safety alert message to RSU_1 while the transmission range is unavailable; CH_3 will trigger for RSU_2 and receive the latest traffic information.

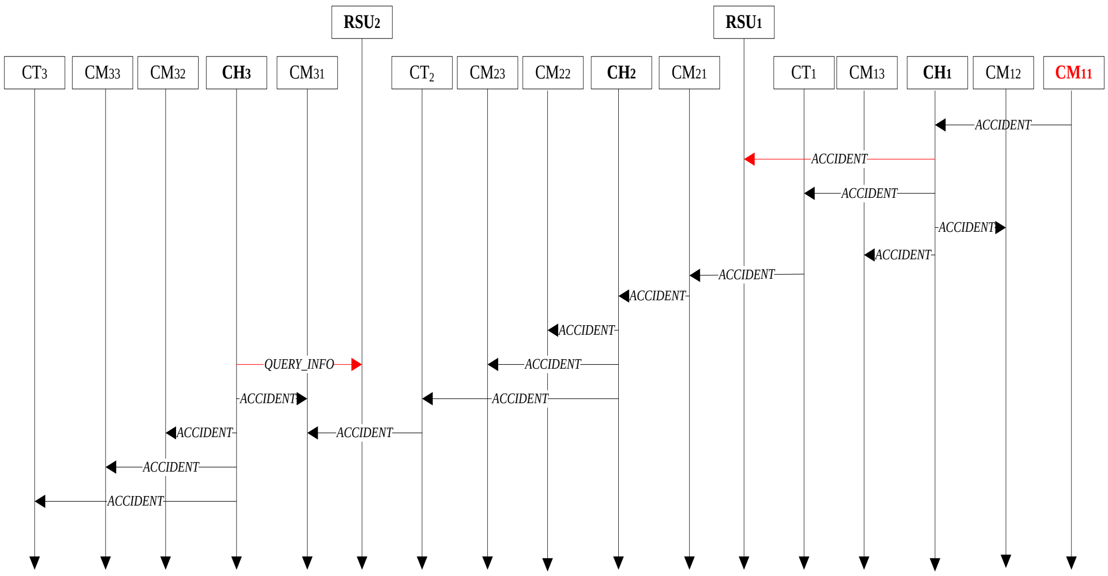

Figure 29: The message dissemination in hybrid mode

In our proposed hybrid message dissemination model, the CH obtains the estimated traffic speed from nearby RSU periodically. When a message such as safety alert send out, the CH will send to all CMs and also send to RSU. When a RSU receive the message from a CH, it will send back to cloud. The updated information on the cloud will also send to related RSUs. All the clusters behind the accident spot will receive alert message from fore cluster and RSU. The message from the RSU will shorten a lot of time for message transfer between clusters. Pseudo Code 4 shows the hybrid message dissemination model.

We will use mathematic analysis to calculate the number of message dissemination and the message deliver time in this proposed mode. There are J clusters to transmit and receive message in the same driving direction. In addition, each vehicle is within the transmission rage of its forwarder. When an accident occurred in the first cluster, Msg is expressed as the total number of this safety alert message dissemination from accident spot to the J_{th} cluster.

The messages send back time from cloud to RSU is defined as Δk. Δk is cluster number between the first CH send message to RSU and first CH receive message from RSU in CN_m.

$$\Delta k = (J - CN_m)$$

The message transferring between CMs who receives message from frontier CT could be described as the following. In cluster *n*, the probability of cluster *n* members receive message from frontier is P_n and total member of cluster *n* is M_n. Every CM receives message will transfer it to its CH. We could express Δm as the message dissemination from CM to the CH.

$$\Delta m = \sum_{1}^{\Delta k} (Pn \ * \ Mn)$$

```
Receive Message:
Switch(Role)
{
Case CH:
        Send Message to CM;
        Send Message to RSU;
        Break;
Case CT:
        If(Message.own == CH)
                Receive message;
        Else
                Transfer received message to hind cluster;
        Break;
Case CM:
        If(Message.own ==CH)
                Receive message;
        Else
                Transfer to CH;
        Break;
Case RSU:
        Send back to cloud;
Default:
        Break;
}
When pass by RSU
{
        If(Role == CH)
                Send request to RSU to query information;
}
```

Pseudo Code 4: The message dissemination model in hybrid mode

Δk also means to be the number that is used CT to transfer message between clusters. The number of the CH that receives message from RSU is Δp.

$$\Delta p = (J - 1 - \Delta k)$$

$$\Delta p = (J - 1 - (J - CN_m))$$

$$\Delta p \text{ could also express as } CN_m - 1$$

RN is the message number that RSU received from cloud. The total message number of the message deliver dissemination could be included: the total CH_i will broadcast in all the J clusters, the message send from CT to other cluster Δk times, The CH gets the message from RSU Δp times, the first CH send one message to RSU to could and the rest CHs get the message from RSU RN times. It could be expressed as below.

$$Msg = J + \Delta \mathrm{m} + \Delta p + RN + 1$$

$$Msg = J + (\sum_{1}^{\Delta k}(\mathrm{Pn} * \mathrm{Mn})) + (CN_m - 1) + RN + 1$$

$$Msg = J + CN_m + RN + \sum_{1}^{\Delta k}(\mathrm{Pn} * \mathrm{Mn})$$

The message transmission time from CM to CH is written as Δt_{m2h}. The total maximum message broadcast time from the CH to all CMs is defined as ΔB. The message transmission time between CT to the hind cluster is written as Δt_{t2c}. The message transmit time between CH to RSU is written as Δt_{h2r}. The message deliver time $D(t)$ would be included: the first message sends out from the accident vehicle to the CH, the CH broadcasts messages to all CMs, Δk times of CT send to the hind cluster (CT to hind cluster ,CM to CH and CH to all CMs), one CH receives message from RSU and broadcast to other clusters.

$$D(t) = \Delta t_{m2h} + \Delta B + \Delta t_{h2r} + \Delta k(\Delta t_{t2c} + \Delta B + \Delta t_{m2h}) + \Delta t_{h2r} + \Delta B$$

$$D(t) = (1 + \Delta k)\Delta t_{m2h} + (2 + \Delta k)\Delta B + 2\Delta t_{h2r}$$

Chapter 6 Simulation Results and Analyses

For VANET simulation, we use the ns-3 and ns3-Highway-Mobility module for our simulations [46],[47],[48]. We evaluate the performances of our proposed message dissemination model and compare with other message dissemination algorithms, included simple broadcast, p-persistence broadcast, slotted p-persistence broadcast, APAL and TLO. We will discuss and compare these methods in message numbers, message collision and message latency.

In Section 6.1, we will describe the assumptions of our simulation. To evaluate the message dissemination effective, we make three assumptions in highway, network and vehicle. In Section 6.2 we will describe the environment of simulation included the detail of the highway and related environment parameters. In Section 6.3, we will list the performance metrics to evaluate our simulations. In Section 6.4, there are four simulation scenarios in our design, total number of message dissemination, message collision, latency in message dissemination and latency between our proposed methods. In these for simulation scenarios, we will improve the proposed V2V and hybrid message dissemination models have better effective. In Section 6.5, we will analyze the simulation results of our proposed message dissemination. With the summarized result, we can find out that our proposed message dissemination models have less message transmitted number and fast dissemination time. At the end, the accuracies of location determination and speed estimation experiments are described in Section 6.6.

6.1 Simulation Assumptions

In our simulation, we simulate a highway environment to evaluate message dissemination performance. To put focus on the message dissemination model, we have some predefine assumption as below.

For highway assumption, every vehicle will move in the same direction and speed of 80 to 110 km/h. Every message dissemination protocol start after accident occurred. There is only one accident occurred in every simulation at the same time. There is no interchange on the highway.

For V2V communications model assumption, all the vehicles are using DSRC cover to transmit message with each other. All the cluster members have routing ability to receive and send for each other.

For V2I communications model assumption, all the vehicles have equipped with device that have ability to communicate to RSUs. All the RSUs receive messages will send back to the cloud server. Every CH will have ability to pull down from the same could server.

6.2 Simulation Environment

Figure 30 illustrates our network environment parameters that defined in a XML for ns-3 simulator. In the simulator, there are three portions of environment setting. The *highwayProject*, *wificonfguration* and *vehicleGenerator* are used to describe the highway environment, wireless connectivity and vehicle parameters, respectively.

We try to evaluate the performance with fairness and realistic environments, the scenario of our message number simulation uses vehicle number as variety. The simulation parameters and their values are summarized in Table 3.

```
<highwayProject numberOfRuns="1" totalTimeInSeconds="30" dt="0.1">
    <highways>
        <highway highwayId="0" numberOfLanes="4" direction="0.0"
        length="1000.0" startX="0.0" startY="5.0" leftTurnSpeed="2.2352"
rightTurnSpeed="2.2352" laneWidth="5.0"/>
    </highways>
    <wifiConfigurations>
        <wifiConfiguration wifiConfigId="1"
wifiStandard="WIFI_PHY_STANDARD_DSRC"
dataMode="OfdmRate6MbpsBW10MHz" txPowerStart="21.5"
txPowerEnd="21.5" txPowerLevels="1" txGain="2.0" rxGain="2.0"
energyDetectionThreshold="-101.0"/>
    </wifiConfigurations>
    <vehicleGenerators>
        <vehicleGenerator highwayId="0" wifiConfigId="1" flow="1.0"
        lowVelocity="60" highVelocity="110" minGap="45.0" VehicleNumber="50"
        penetrationRate="100.0" ClusterNumber="5"
messageDisminstaion="Collaborator_message_disseminstaion" />
    </vehicleGenerators>
</highwayProject>
```


Figure 30: Environment setting file for ns-3 simulator

Table 3: Simulation parameters and their values

Environment Parameters	Value
Highway Lane	4
Highway Distance	100 KM
Highest Velocity	110 km/h
Lowest Velocity	60 km/h
Transmission Range	1000 M
Wireless Interface	DSRC
Communication Mode	Proposed V2V, Proposed Hybrid
Number of Vehicle	10,20,30,40,50,60,70,80,90,100
Nodes in Cluster	10

6.3 Performance Metrics

For a message dissemination protocol, we take the total number of message dissemination to be the first metric of performance. The number of dissemination message plays an important role. When message disseminating, the more messages mean more chance to collision and lost. Especially in dynamic network environments, the more messages will make more loads in message dissemination. Second, the message collision is our second metric of performance. The more message collision on the way of dissemination, it will make message dissemination need extra cost to retransmit the broking message. A well performance message dissemination protocol keeps the message collision in much low chance to reduce the extra resend message cost. The last metric of performance is the message latency. For an accident occurred, a safety alert message will be disseminated to on-coming vehicles to avoid traffic congestions. The message latency indeed plays very important role in VANETs. The real-time traffic information can be disseminated in a rapidly and effective way will be feasible for driving safety and transportation efficiency applications in VANETs.

6.4 Simulation Cases Design

To evaluate the total number of message dissemination, message collision and latency message dissemination, we assume that there will be an accident occurred in the highway. The accident car will send out message to other nodes. Every message will be disseminate in different message dissemination algorithms, included simple broadcast, P-persistence, Slotted p-persistence, TLO, APAL, the proposed V2V and the proposed hybrid approach.

In addition, we will simulate the message disseminate with different vehicle numbers from 10 to 100. With the simulation case, we will record the total message number, message collision and latency during message dissemination process. There are different distances between the latency of our proposed V2V and hybrid message dissemination models.

6.5 Results and Analyses

The main object of our proposed hybrid message dissemination model is to reduce the number of message dissemination and improve the efficiency. The following simulation environment is in the regular highway traffic regime. There are four simulations to measure the performance of the message dissemination, including total number of message dissemination, message collision, message latency and two proposed methods.

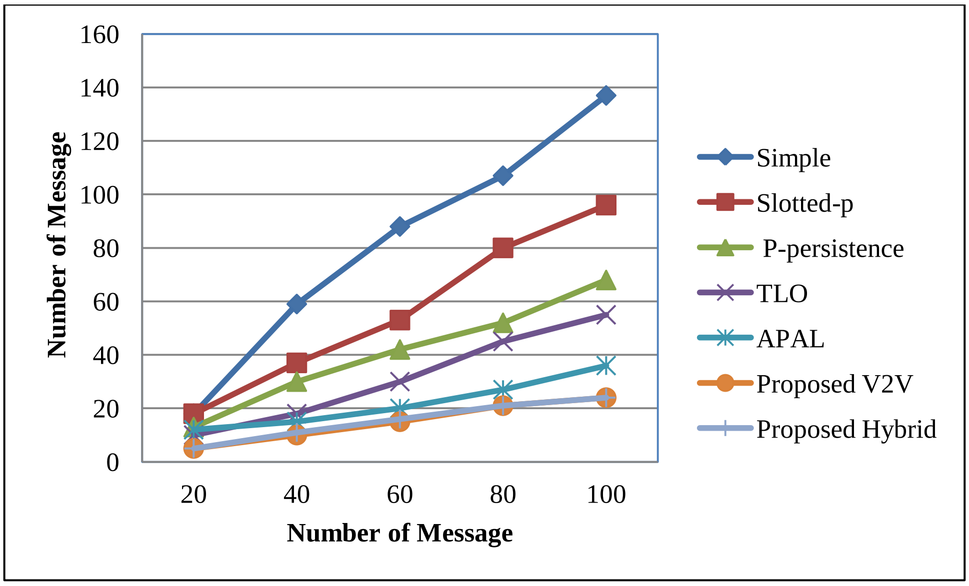

Figure 31: Total number of message dissemination

Figure 31 shows the total amount of messages disseminated in our cluster-based message dissemination protocol could be reduce. Our proposed protocol has better result than previous best APAL protocol. With our effective design, the message number would not be have burst grow when the number of vehicles are increasing. In addition, both of our V2V and hybrid methods have almost the message dissemination number.

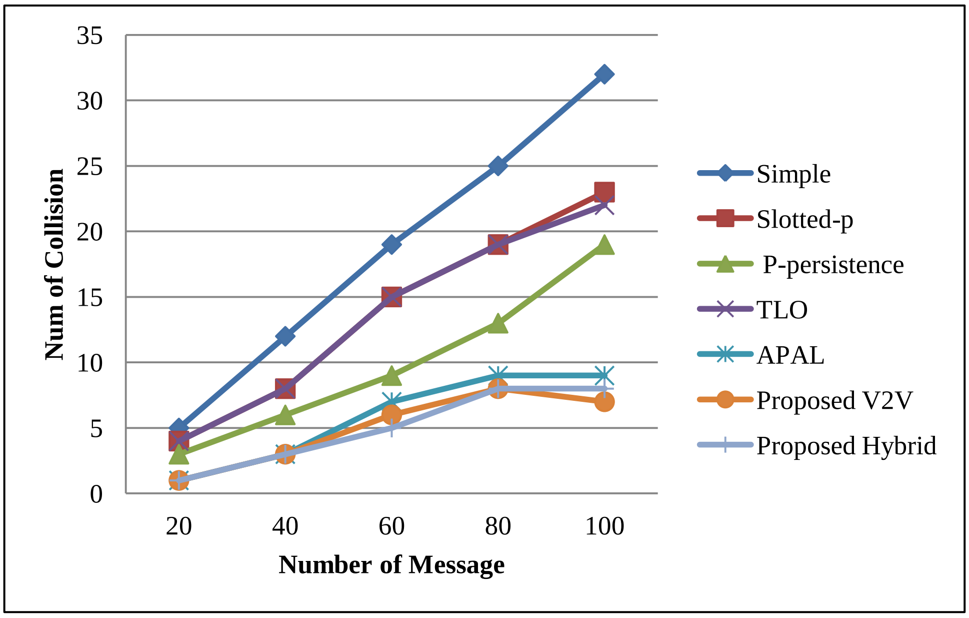

Figure 32: Message collision

Figure 32 shows the number of message collision in message dissemination process. More messages transmitted might have more chances to occur message collisions. In our clustering design, it will reduce the message collision problem. The main reason is that our proposed approach has relations with two special roles: CH and CT. The CH is responsible for holding its cluster and disseminating message to the cluster members; and CT is responsible for delivering message to hind cluster which is the furthest from CH in the following direction. Therefore, the proposed approach will reduce the broadcast or rebroadcast collision problems.

We can observe the comparison between the latency time and the number of message propagation with different methods in Figure 33. It is clear that, our proposed hybrid message dissemination approach has less latency time. The reason is that all the CHs that receive message from the RSUs would obtain the real-time information. The message exchange between CH and RSU could reduce the inter-cluster message transferring time and downgrade the latency problem.

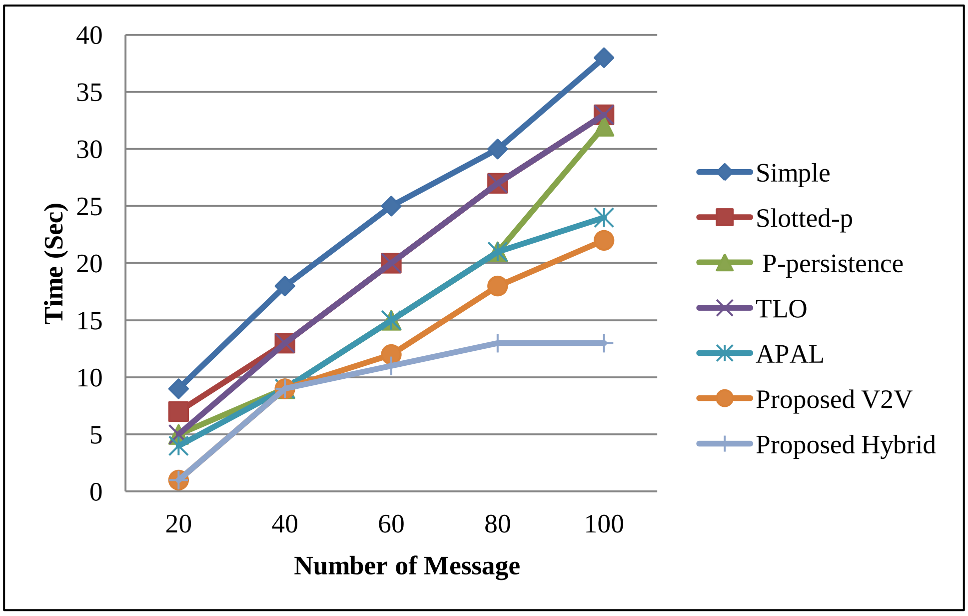

Figure 33: The latency in message dissemination

Figure 34 shows the comparison between the latency time and the number of message dissemination with our proposed V2V and the proposed hybrid approach. The result shows that the proposed hybrid approach has less latency than the proposed V2V while the number of message is higher than 60. The hybrid design (V2V+ V2I) is helpful and useful to message dissemination and with better efficiency.

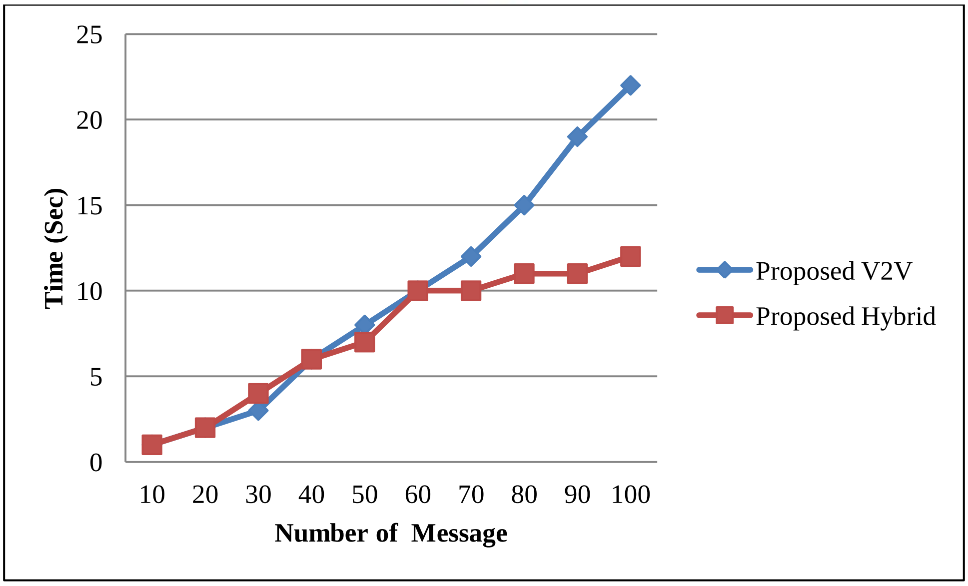

Figure 34: The latency in message dissemination between the proposed methods

6.6 The Performance of Cellular-based Traffic Data Estimation

The aim of the experiments is to analyze the accuracies of location determination and speed estimation. The experiments are carried out on a 12.2 km long expressway segment in the Expressway No. 66 in Taiwan. The segment is driven 12 times with an MS device (e.g., HTC Desire running Android platform 2.2) with a GPS receiver. Signaling data and geographical coordinates are collocated from a MS simultaneously. In our experiments, there are 96 different BSs detected on the expressway segment.

For location determination, the accuracy of location determination by using FPA is assessed using data from 12 test runs to evaluate each location. There are 6299 records are collected from the MS in test runs. We measure the performance of our approach in the way called k-fold cross-validation [49]. In experiments, the training and testing are performed for 6299 locations. In iteration i, location l_i is selected as the test corpus, and the other locations are collectively used to be training locations. In these experiments, the feasibility of using the FPA and classical cell ID location method [50] is evaluated.

Table 4 shows the experiment results of location determination with FPA algorithm and cell ID method in Expressway No. 66 in Taiwan. Consider FPA first; it can be observed that the average errors of location determination are about 38.46 meters, 36.11 meters, and 37.14 meters when the different numbers of the nearest locations (k) are from 1 to 3. Second, we consider cell ID based location method [51] to implement and evaluate that the errors of location determination is about 714.07 meters.

For speed estimation, we collect two consecutive determinate locations by analyzing cellular network signal and use Eq. (4.4) to calculate the estimated speed (u_e). In experiments, we collect the real speed (u_r) information obtained from GPS receiver and analyze the error ratio (E) of speed estimation by Eq. (6.1). FPA can be observed that the average error ratios of speed estimation are about 3.95%, 3.39%, and 4.52% when the different numbers of the nearest locations (k) are from 1 to 3. Second, we implement the cell ID based location method and handover based location method. Then we evaluate that the error ratios of speed estimation using the two location methods are about 85.32% and 77.98, respectively. The results show that the accuracy is quite good for a majority of the speed estimation by using FPA with the nearest locations (k) as 2.

$$E = \min\left(\frac{|u_r - u_e|}{u_r}, 1\right) \times 100\% \tag{6.1}$$

Table 4: The experiment results of location determination with different algorithms

Location Method	Average Error of Location Determination(meter)
FPA with k = 1	38.46
FPA with k = 2	**36.11**
FPA with k = 3	37.14
Cell ID based location method	714.07

The advantages of our proposed method are as follows: (1) FPA is completely done by the back-end server without the needs of additional modification of mobile terminal. (2) FPA can perform the estimation in any location rather than the handover based location method can only be conducted in the handover zone. However, the disadvantage of our method is that it will generate a large amount of training data and takes considerable time to calculate. We have adopted MapReduce algorithm to accelerate the computations in this article.

Chapter 7 Conclusions and Future Work

The main contribution of this work is that it presents a hybrid (V2V and V2I) message dissemination architecture to transfer real-time traffic information. In V2V mode, we propose a consensus leader election algorithm and cluster-based message dissemination model. The trade-off of the proposed message dissemination architecture is the CH and the CT roles designated. The CH is responsible for holding its cluster and disseminating message to the cluster members; and CT is responsible for delivering message to hind cluster. With our design, the proposed approach will reduce the broadcast or rebroadcast collision problems. In V2I mode, we propose a novel speed estimation method based on FPA algorithm. In the hybrid communications mode, we use the proposed cluster-based message dissemination model and cloud computing to fasten the message dissemination time. With RSU and cloud server, the CH can query the real-time information from cloud directly. In simulations, we compare our proposed message dissemination methods with other message dissemination protocol in VANETs. The results show that our proposed message dissemination architecture not only disseminates fewer messages but also has shorter deliver time than other message dissemination protocols.

Future study could be pursed along with several of the following directions:

1. The proposed consensus cluster head election algorithm not only reduces the number of message passing but also improves network consistency. We will apply the proposed election algorithm to other different fields, such as social network connection and resources management and wireless P2P network group formation.
2. The current growth in cloud computing is creating great demand for calculating large-scale data. Future study may focus on using cell phones as probes to obtain

more traffic information and calculate in cloud. To develop real-time relations with neighboring clusters/members is to feast the traffic improvement.

3. In this research, we only discuss the one way direction message dissemination architecture. The dual directions message dissemination architecture will be designed to investigate.

References

[1] M. L. Sichitiu and M. Kihl, “Inter-Vehicle Communication Systems: A Survey,” *IEEE Communications Surveys & Tutorials*, vol.10 no.2, pp.88-105, 2008.

[2] T. L. Willke, P. Tientrakool, and N. F. Maxemchuk, “A Survey of Inter-Vehicle Communication Protocols and Their Applications,” *IEEE Communications Surveys & Tutorials*, vol. 11, no. 2, pp.3-20, 2009.

[3] P. Papadimitratos “Vehicular Communication Systems: Enabling Technologies, Applications, and Future Outlook on Intelligent Transportation,” *IEEE Communications Magazine*, pp.84-95, 2009.

[4] A. M. Vegni and T. D. C. Little, “Hybrid Vehicular Communications Based on V2V-V2I Protocol Switching,” *Int. Journal of Vehicle Information and Communication Systems*, vol.2, no.3/4, pp.213-231, 2011.

[5] M. Rathod, “A hybrid data dissemination scheme for VANETs,” Wireless Day, pp.1-7, 2011.

[6] T. Nadeem, “A Comparative Study of Data Dissemination Method for VANETs,” *Third Annual International Conference on Mobile and Ubiquitous Systems: Networking & Services*, Jul. 2006.

[7] N. Caceres, J.P. Wideberg, and F.G. Benitez, “Review of Traffic Data Estimations Extracted from Cellular Networks,” *IET Intelligent Transport Systems*, vol.2, no.3, pp.179-192, 2008.

[8] O. Tonguz, N. Wisitpongphan, F. Bai, P. Mudalige and V. Sadekar, “Broadcasting in VANET,” in *Proc. IEEE Conf. on Mobile Networking for Vehicular Environments*, pp.7-12, 2007.

[9] N. Wisitpongphan, O. Tonguz, J. Parikh, F. Bai, P. Mudalige, and V. Sadekar, “On the Broadcast Storm Problem in Ad hoc Wireless Network,” in *IEEE Wireless Communications*, to appear.

[10] O. Tonguz, N. Wisitpongphan, F. Bai, P. Mudalige and V. Sadekar, “Broadcast Storm Mitigation Techniques in Vehicular Ad Hoc Networks,” *IEEE Wireless Communications*, pp.84-94, 2006.

[11] K. Suriyapaibonwattana and C. Pomavalai, "An Effective Safety Alert Broadcast Algorithm for VANET," *International Symposium on Communications and Information Technologies, 2008.*

[12] K. Suriyapaibonwattana, C. Pomavalai and G. Chakraborty, "An Adaptive Alert Message Dissemination Protocol for VANET to Improve Road Safety," *IEEE International Conference on Fuzzy Systems*, Aug. 2009.

[13] P. Fan, "Improving Broadcasting Performance by Clustering with Stability for Inter-Vehicle Communication", *Proceeding of IEEE 65th VTC2007- Spring Vehicular Technology Conference*, pp. 2491-2495, Apr. 2007.

[14] T. D. C. Little and A. Agarwal, "An Information Propagation Scheme for VANETs," in *Proc. IEEE Conf. on Intelligent Transportation Systems*, pp.155-160, 2005.

[15] C. Caballero-Gil, P. Caballero-Gil and J. Molina-Gil, "Using Groups to Reduce Communication Overhead in VANETs," *The Second International Conference on Advances in P2P Systems*, Florence, Italy, Oct. 2010.

[16] D. Wei, H. A. Chan, "A Survey on Cluster Schemes in Ad Hoc Wireless Networks," *IEE Mobility Conference*, Guangzhou, China, Nov. 2005.

[17] E. Souza, I. Nikolaidis and P. Gburzynski, "A New Aggregate Local Mobility Clustering Algorithm for VANETs," http://www.olsonet.com/pg/PAPERS/icc10.pdf

[18] M. Raya, A. Aziz and J.P. Hubaux, "Efficient Secure Aggregation in VANETs," *Proceedings of the 3rd international workshop on Vehicular ad hoc networks*, Los Angeles, USA, Sep. 2006.

[19] H. Su and X. Zhang, "Clustering-Based Multichannel MAC Protocols for QoS Provisionings Over Vehicular Ad Hoc Networks," *IEEE Transactions on Vehicular Technology*, vol. 56, no 6, 2007.

[20] Y.C. Chu and N.F. Huang, "An Efficient Traffic Information Forwarding Solution for Vehicle Safety Communications on Highways," *IEEE Transactions on Intelligent Transportation Systems*, pp.1-13, 2011. (Early Access Articles)

[21] W. Kim, "AUDITOR: A framework for high availability of DB/DC systems," IBM Res. Rep. RJ3512, 1982.

[22] H.Garcia-Molina, “Elections in a Distributed Computer System,” *IEEE Transactions on Computers*, vol. c31, no. 1, pp.48-59, 1982.

[23] N. Fredrickson and N. Lynch, “Electing a Leader in a Synchronous Ring,” *J.ACM*, vol.34, no.1, pp.98-115, 1987.

[24] S. H. Park, Y. Kim, and J. S. Hwang, “An efficient algorithm for leader-election in synchronous distributed systems,” *IEEE TENCON*, pp.1091-1094, 1999.

[25] M. M. Rahman and A. Nahar, “Modified Bully Algorithm using Election Commission,” Available at: http://arxiv.org/ftp/arxiv/papers/1010/1010.1812.pdf

[26] M. Gholipour, M. S. Kordafshari, M. Jahanshahi, A. M. Rahmani, “A new approach for election algorithm in distributed systems” 2009 Second International Conference on Communication Theory, Reliability, and Quality of Service, Colmar, France, Jul. 2009.

[27] L. Lamport, “The part-time parliament,” *ACM Transactions on Computer Systems*, vol. 16, no. 2, pp.133-169, 1998.

[28] L. Lamport, “Paxos made simple,” Nov. 2001.
Available at: http://pdos.csail.mit.edu/6.824/papers/paxos-simple.pdf

[29] H. C. Chang and C.C. Lo, “A Consensus-Based Leader Election Algorithm for Wireless Ad Hoc Networks,” *Proceedings of 2012 International Symposium on Computer, Consumer and Control*, Taichung, Taiwan, Jun. 2012.

[30] S. Logghe and S. Maerivoet, “Validation of Travel Times Based on Cellular Floating Vehicle Data,” *Proceedings of the 6th European Congress and Exhibition on Intelligent Transportation Systems*, Aalborg, Denmark, 2007.

[31] L.M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, “A Break in the Clouds: Towards a Cloud Definition,” ACM SIGCOMM Computer Comm. Rev. vol. 39, no. 1, pp. 50–55, 2009.

[32] M. Armbrust, et al., “A View of Cloud Computing,” Communications of the ACM, vol. 53, no.4, pp.50-58, 2010.

[33] P. Mell, T. Grance, The NIST Definition of Cloud Computing, Version 15, 2009.

[34] I. Foster, Y. Zhao, I. Raicu, and S. Lu, “Cloud Computing and Grid Computing 360-degree Compared,” *Grid Computing Environments Workshop*, Austin, USA, Nov. 2008.

[35] J. Dean and S. Ghemawat, “Mapreduce: Simplified Data Processing on Large Clusters,” *Communication of ACM*, vol. 51, no. 1, pp. 107-113, 2008.

[36] M. Zaharia, A. Konwinski, A. D. Joseph, R. Katz and I. Stoica, "Improving MapReduce Performance in Heterogeneous Environments," *Proceedings of the 8th USENIX Conference on Operating Systems Design and Implementation* (OSDI'08), pp.29-42, 2008.

[37] S. Ghemawat, H. Gobioff and S. T. Leung, "The Google File System,"*SOSP'03*, 2003.

[38] Hadoop Distributed File System. Available at: http://hadoop.apache.org/hdfs/

[39] K. Mavlonov, K. Hirata, Y. Higami, S.Y. Kobayashi, "Auxons: A Large Scale Distributed Storage System for Semi-structured Data", in *Proc. First Russia and Pacific Conference on Computer Technology and Applications*, pp. 120-124, 2010.

[40] C.H. Chen, B.Y. Lin, H.C. Chang, and C.C. Lo, "The Novel Positioning Algorithm Based on Cloud Computing - A Case Study of Intelligent Transportation Systems," *Information-An International Interdisciplinary Journal*, In Press, Accepted Manuscript, 2011.

[41] S. Hussain and M. S. Rahman, "Using Received Signal Strength Indicator to Detect Node Replacement and Replication Attacks in Wireless Sensor Networks", *Proc. SPIE*, Orlando, USA, Apr. 2009.

[42] B.Y. Lin, "A Traffic Speed Estimation Mechanism Based on the Signals of Cellular Networks," National Chiao-Tung University, Doctor Dissertation, 2011.

[43] B. Y. Lin, C. H. Chen, C. C. Lo, "A Novel Speed Estimation Method Using Location Service Events Based on Fingerprint Positioning," *Advanced Science Letters*, vol.4, no.11-12, pp.3735-3739, 2011.

[44] C. H. Chen, B. Y. Lin, C. H. Lin, Y. S Liu, C. C. Lo, "A Green Positioning Algorithm for Campus Guidance System," *International Journal of Mobile Communications*, vol. 10, no. 2, pp. 119-131, 2012.

[45] S. A. Dudani, "The Distance-Weighted k-Nearest-Neighbor Rule," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 6, no. 4, pp. 325-327, 1976.

[46] H. Arbabi and M. C. Weigle, "Highway Mobility and Vehicular Ad-Hoc Networks in NS-3," *Proceedings of the 2010 Winter Simulation Conference*, 2010.

[47] ns-3. Available at: http://www.nsnam.org/

[48] ns-3-Highway-Mobility.
Available at: http://code.google.com/p/ns-3-highway-mobility/

[49] J. Han and M. Kamber, "Data Mining: Concepts and Techniques," second edition, Morgan Kaufmann Publishers, 2005.

[50] 3GPP Technical Specification Group (TSG) Services and System Aspects, "Location Services (LCS); Service description; Stage 1 (Release 9)", TS 22.071, version 9.1.0, 2010.

[51] Bar-Gera, H., "Evaluation of a Cellular Phone-based System for Measurements of Traffic Speeds and Travel Times: a Case Study from Israel," *Transportation Research C*, vol. 15, no. 6, pp. 380–391, 2007.