The K-hop Cooperative Video Streaming Protocol Using H.264/SVC Over the Hybrid Vehicular Networks

Chao-Hsien Lee, Chung-Ming Huang, *Senior Member, IEEE,* Chia-Ching Yang, and Hsiao-Yu Lin

Abstract—When a number of persons, e.g., a family or a group of friends, drive their vehicles for a trip together, they can form a fleet of vehicles and share their network resources during their trip. Let one member want to watch a video from the Internet. He may not have high resolution or video quality due to his limited 3G/3.5G bandwidth to the Internet. The cooperative video streaming scenario allows the requested member to ask other members of the same fleet to download video cooperatively. In other words, other members can help to download parts of the video from the Internet and then forward video data to the requested member hop by hop through the ad-hoc network. This work proposes the k-hop cooperative video streaming protocol using H.264/SVC over the hybrid vehicular networks which consist of 3G/3.5G cellular network and Dedicated Short-Range Communications (DSRC) ad-hoc network. In order to smooth video playback over the DSRC-based ad-hoc network, this work proposes: (1) one streaming task assignment scheme that schedules the streaming task to each member over the dynamic vehicular networks, and (2) packet forwarding strategies that decide the forwarding sequence of the buffered video data to the requested member hop by hop. Finally, we utilize the network simulator version 2 (NS2) to simulate the proposed protocol. Based on the simulation results, the proposed scheme can estimate the assignment interval adaptively and the playback priority first (PPF) strategy has the best performance for the k-hop video forwarding over the hybrid vehicular networks.

Index Terms—Cooperative streaming, scalable video coding, vehicular ad-hoc network (VANET), dedicated short-range communication (DSRC)

✦

1 INTRODUCTION

Due to maturity of multimedia processing and network technologies, i.e., H.264/SVC codec and 3G/3.5G/4G wireless network, the demands of universal multimedia service are increased significantly. A practical example is that a family or a group of friends drive their sedans or RVs to take a road trip together, e.g., from Los Angeles to Las Vegas in U.S.A. or from Paris to Amsterdam in Europe. They drive their vehicles to meet in an entrance point of the highway and then form a fleet along the highway. In other words, a fleet composed of several vehicles starts from the same point and has the same travelling route and the same destination. If one of the vehicles in a fleet wants to request a video stream from the Internet, it can download video data using 3G/3.5G cellular network. Since the bandwidth of the 3G/3.5G network over the moving vehicular networks is unstable and insufficient, the video quality of the requested video stream may not be good enough. Even using 4G network, the bandwidth still may not be enough for the following concerns. First, other applications may utilize the 4G network simultaneously. Second, the moving behavior of one vehicle, e.g., moving with high speed or around the coverage boundary of one base station, makes the decaying of 4G bandwidth. In order to increase the video quality during the travelling path, one vehicle would ask other vehicles belonging to the same fleet to download video data using their redundant 3G/3.5G bandwidth[1]. Once other vehicles download video data from the Internet, they forward the downloaded video data to the requested vehicle through the ad-hoc transmission among vehicles, in which Dedicated Short-Range Communications (DSRC) is designed for automotive to have one-way or two-way short to medium-range wireless communication specifically in the highly dynamic mobile environment. In this work, the aforementioned scenario is defined as Cooperative Video Streaming (CVS) over the hybrid vehicular networks, which consists of (1) 3G/3.5G network for vehicle-to-infrastructure (V2I) communication and (2) DSRC ad-hoc network for vehicle-to-vehicle (V2V) communication.

Three critical roles of the proposed CVS are (1) requester, (2) forwarder, and (3) helper. Fig. 1 depicts the proposed

---

- *C.-H. Lee is with the Department of Electronic Engineering, National Taipei University of Technology, Taipei 10608, Taiwan. E-mail: chlee@mail.ntut.edu.tw.*
- *C.-M. Huang, C.-C. Yang, and H.-Y. Lin are with the Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan 701, Taiwan. E-mail: {huangcm, yangcc, sylin}@locust.csie.ncku.edu.tw.*

Manuscript received 26 Oct. 2012; revised 1 Nov. 2013; accepted 14 Nov. 2013. Date of publication 4 Dec. 2013; date of current version 29 May 2014. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier 10.1109/TMC.2013.153

[1] Subscribers of 3G/3.5G network in some countries, e.g., Taiwan, can subscribe 3G/3.5G service with a fixed monthly payment for some specific transmission rate and the use of the 3G/3.5G network is unlimited. In other words, subscribers can have Internet service as long as they want with a fixed cost per month over 3G/3.5G network. Thus, users of the same group are straightforwardly assumed to be willing to share their 3G/3.5G bandwidth with each other.

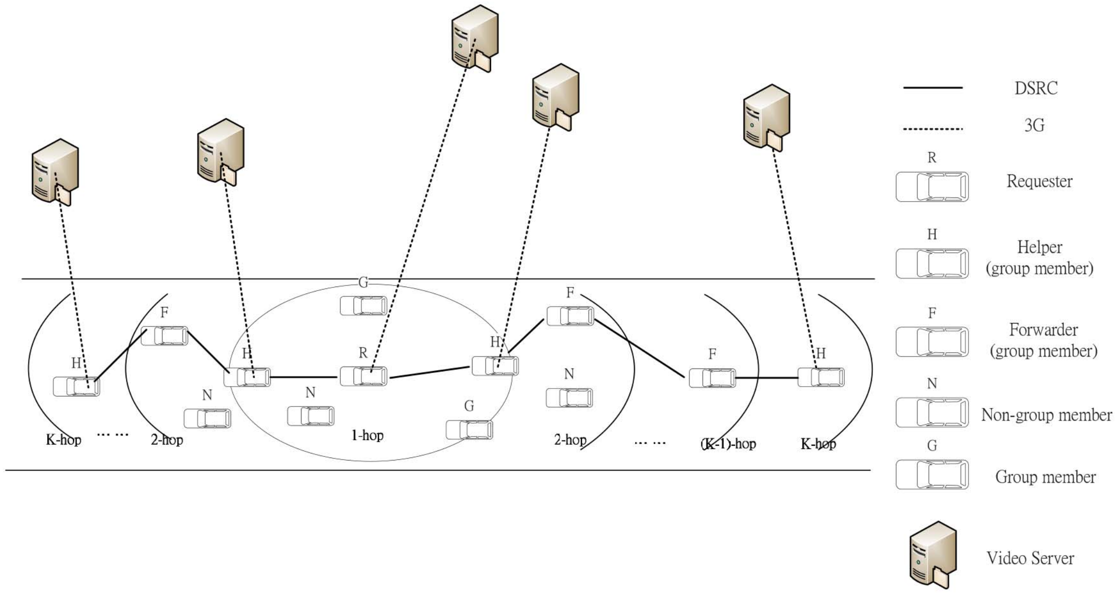


Fig. 1. Illustrated scenario of the proposed k-hop CVS protocol.

CVS scenario in this paper. A requester is the member having the demand for video streaming service. Forwarders are members that are responsible for forwarding pieces of video data hop by hop through DSRC-based ad-hoc network. Helpers are members that not only forward video data through DSRC-based ad-hoc network but also utilize their 3G/3.5G interfaces to download video data from the Internet. In our assumption, these members know each other in advance, e.g., they may be friends or families. However, referring to Fig. 1, the vehicular networks' condition is dynamic and unpredictable. Some vehicles that do not belong to the fleet would jam the highway and then let the distance among members be more than one hop. Thus, the downloaded video data should be forwarded hop by hop and may be transmitted from multiple paths back to the requester. This work proposes the k-hop CVS protocol which is adaptive to the hybrid vehicular networks environment. This work studies and resolves the CVS scenario in the application layer. Problems and issues of lower layers, i.e., transport layer, network layer and link layer, are beyond the scope of this work.

In the proposed CVS protocol, four issues that need to be resolved for having cooperative video streaming are (1) helper selection, (2) video packetizing, (3) streaming task assignment and (4) packet forwarding strategy. The first two issues, i.e., (1) helper selection—how to select helpers from neighboring members and (2) video packetizing— how to packetize layered video data for delivery, have essentially been studied in our preliminary research [22],[23]. The other two issues, i.e., (3) streaming task assignment—how to assign streaming tasks to helpers and forwarders and (4) packet forwarding strategy—how to decide the forwarding sequence of the buffered video data in a helper or forwarder to the requester, are resolved in this work. First, this work adopts the scalable video coding technique, i.e., H.264/SVC, to encode video into one base layer and multiple enhancement layers. The base layer is downloaded by the requester itself. On the other hand, enhancement layers are transmitted through DSRC-based ad-hoc network that is composed of helpers and forwarders. Due to possible traffic conditions in reality, each helper's 3G/3.5G bandwidth and the hop-count distance between each helper and the requester may change with time. Hence, the proposed k-hop CVS protocol includes a streaming task assignment scheme which adapts the assignment interval according to helpers' 3G/3.5G bandwidth and decides the task scheduling sequence based on the playback time of video data and the hop-count distance between helpers and the requester. Second, since enhancement layers are downloaded by helpers, enhancement layers need to be passed through the DSRC-based ad-hoc network. For a forwarder, it may receive video data from different helpers or forwarders simultaneously. Since the DSRC network bandwidth is still limited, which buffered video data in a forwarder should be transmitted first would affect the video quality of the requester. Therefore, the proposed k-hop CVS protocol considers (1) arrival time, (2) playback priority and (3) available bandwidth individually. Several transmission strategies inside a forwarder for forwarding video data over the DSRC-based ad-hoc network are discussed and analyzed in this work.

The rest of this paper is organized as follows. Section 2 presents related works about the CVS. Section 3 reviews our preliminary study, including helper selection and video packetizing schemes. Section 4 describes the proposed streaming assignment scheme and Section 5 describes the proposed packet forwarding strategies in details. Section 6 demonstrates the simulation results. Finally, we summarize and conclude our work in Section 7.

2 RELATED WORKS

In this section, we survey existing research issues in the cooperative video streaming, including (1) neighbor discovery over vehicular ad-hoc network, (2) cooperative network integrating WWAN and WLAN, (3) multipath routing and bandwidth aggregation, and (4) scalable video coding.

2.1 Neighbors Discovery Over Vehicular Ad-Hoc Network

How to discover neighboring vehicles quickly and efficiently is one of the critical issues in vehicular ad-hoc network (VANET). Regarding group formation in VANET, Luo and Guo utilized a CDMA-like on-off signalling for group testing and then the received node can infer which nodes are neighboring based on the overall on-off pattern [2]. Tsai *et al*. proposed a car fleet maintenance scheme based on a swarming model and the small world phenomenon [3]. In [3], a dynamic clustering protocol classifies vehicles into pseudo-leaders and followers. According to traffic and road conditions, pseudo-leaders would determine their positions within the fleet and then decide to speed up or slow down for fleet cohesion and formation. Khalili *et al*. introduced (i) an energy detection mechanism that enables nodes to estimate their reception status and (ii) a feedback mechanism that provides collision information to the transmitters [4]. According to the simulation and analysis results, reception status feedback can reduce the time of the neighbor discovery process.

Once neighboring vehicles can form a group, they can improve network performance by sharing network resources. Taleb *et al*. surveyed a routing protocol to guarantee the communication stability of VANET [5]. In [5] the proposed protocol groups vehicles based on their movement information, e.g., position, speed and direction. Okamura *et al*. proposed wireless direct distribution protocol (WDDP) to provide a group communication mechanism among vehicles [6]. Two phases in the WDDP are (1) group construction phase, which lets vehicles form a group and (2) consensus formation phase, which enables request and data exchange among vehicles. Zoican and Galatchi implemented a predictable routing protocol (PRP) [7]. In [7], the proposed PRP groups vehicles based on velocity vectors and analyzes the possibility to predict a link breakage event before its occurrence. Then, the PRP would determine another more stable route. The aforementioned grouping mechanisms considered how to construct a group to disseminate data. In this paper, our proposed scheme concentrates on how to have group formation to aggregate bandwidth from tightly-coupled group's members, which travel on the same route to the same destination, not only for forwarding data in the ad-hoc paths but also for downloading data from 3G/3.5G networks to have video streaming.

2.2 Cooperative Network Integrating WWAN and WLAN

There were some researches concentrating on the idea of cooperative downloading. In [8], the authors proposed a cooperative strategy for content delivery and sharing in vehicular networks, in which the proposed strategy did not focus on streaming and is designed to gather part of the data from 1-hop helpers only. In [9], authors proposed a system for mobile devices that receive the same video stream and thus can share received video data over WLAN. In their research, they designed a distributed leader election algorithm for the cooperative system, and analytically showed that the proposed system is outstanding in terms of energy consumption and channel switching delay. However, in the VANET environment, the considered mobile nodes are unlike the traditional mobile devices in two aspects: (1) the power consumption is no longer the main issue and (2) the computing capability is more powerful to run complicated tasks. Regarding the cooperative streaming scenario, a collaborative downloading system called COMBINE was designed by Ananthanarayanan *et al*. [10]. COMBINE integrates neighboring nodes' Wireless Wide Area Network (WWAN) interfaces to download resources for an active node. Then, neighboring nodes deliver data to the active node using their Wireless Local Area Network (WLAN) links. Furthermore, the cooperative streaming scenario may adopt different codecs to encode video data. For example, Leung and Chen proposed a protocol called Collaborative Streaming among Mobiles (COSMOS) using the MDC codec in wireless networks [11]. On the other hand, Fan *et al*. described a joint session scheduling (JOSCH) mechanism using layer-encoded streaming in heterogeneous wireless networks [12].

In [24], Guan *et al*. proposed a cross layer scheme using rate control, relay selection and power control for video streaming. However, their proposed system considered single hop network and the scenario is for multimedia sensor network. Our proposed system considers multiple hop network and the corresponding scenario is for the vehicular ad-hoc network. Thus, the considered technical issues are different. In [25], Xing *et al*. proposed a cooperative vehicle relay scheme using vehicles and road side unit to enhance video quality based on the variation of network condition. The main differences between the work depicted in [25] and our work are as follows: (1) instead of using road side unit, we consider 3G/3.5G network to enlarge the wireless signal coverage for downloading video; (2) the selected helpers are chosen to aggregate multiple nodes' wireless network bandwidth to increase downloaded throughput based on their 3G/3.5G and DSRC networks in our work, in comparison, they are only for wireless network relay in the work of [25].

2.3 Multipath Routing and Bandwidth Aggregation

A related issue of the CVS system is the multipath routing problem. Except those 1-hop cooperative helpers, a helper may use multiple routing paths to send the data to the requester. In [13], the authors presented an architecture supporting transmission of multiple video streams in ad-hoc networks by establishing multiple routing paths to provide extra video coding and transport schemes. In [14], the proposed multipath transmission control scheme not only aggregates the available bandwidth of multiple paths, but also reduces the unnecessary time of packet reordering at the receiver. In [15], authors proposed a protocol that selects multiple maximally disjointed paths without causing flow congestion. Although a lot of researches have addressed the problem of multipath routing, most of them concentrated on how to find paths providing good quality to send data back, but how to find appropriate cooperative helpers is left without answers.

Bandwidth aggregation is the basic and fundamental issue of the CVS system. Some papers have addressed this issue. In [19], authors concentrated on the bandwidth aggregation of a host by simultaneously using multiple interfaces and presented a network layer architecture that enables diverse multi-access services. In [20], the authors proposed a k-path proxy discovery algorithm to aggregate idle cellular links' bandwidth of others, which is similar to our work. However, the proposed k-path proxy discovery algorithm doesn't consider issues of the highly mobile environment in VANET and the cooperative streaming applications. In [21], authors concentrated on the bandwidth aggregation service for an Internet "aggregation proxy". The proxy allocates channels from participants in MANET and strips packets among them, and participants then forward packets to the destination in MANET. In the proposed architecture, however, the scalability problem may be the most obstacles because all additional functionalities are constructed on proxies.

3 REVIEW OF OUR PRELIMINARY STUDY

As described in Section 1, we have resolved the first two issues in our preliminary studies [22],[23]. The brief review of our previous studies in [22] and [23] is given in this section. In order to provide higher quality of video to the requester, how to select suitable members as helpers to download video from the Internet is the main challenge tackled in [22]. The Greedy APproach (GAP) is designed to achieve the maximal throughput in the CVS scenario. In the proposed scheme, members would reply their (1) hop-count distance, (2) available bandwidth of 3G/3.5G and DSRC, and (3) surrounding members to the requester. The requester executes the greedy approach (GAP) to select suitable members as helpers. The basic idea of each method is firstly to rank members according to their capacities in both 3G/3.5G and DSRC interfaces and the hop-count distance from their replied messages. Then, each method will verify candidate helpers from the highest ranking one and check whether adding the new traffic flow generated by selecting the member in the i-th hop as a helper would exceed left DSRC available bandwidth of any selected helper in the (i-1)-th hop or not.

An example of using the proposed GAP is depicted in Fig. 2. Let there be a requester and six members A, B, C, D, E, and F. The solid line connecting two vehicles represents that these two vehicles are within each other's DSRC coverage range. On the other hand, the dashed line represents the 3G/3.5G cellular link. Let the requester require 600k bandwidth and be trying to select some members to be helpers to share their redundant bandwidth. At the beginning of the proposed scheme, the requester will classify these members into different hop-count groups based on their hop counts, e.g., A and B are in the 1-hop group, C and D are in the 2-hop group, E and F are in the 3-hop group.The requester has the following result after ranking each hop-count group: 1-hop = {B, A}, 2-hop = {C, D}, and 3-hop = {E, F}. After that, the requester attempts to select helpers from 1-hop to 3-hop orderly.

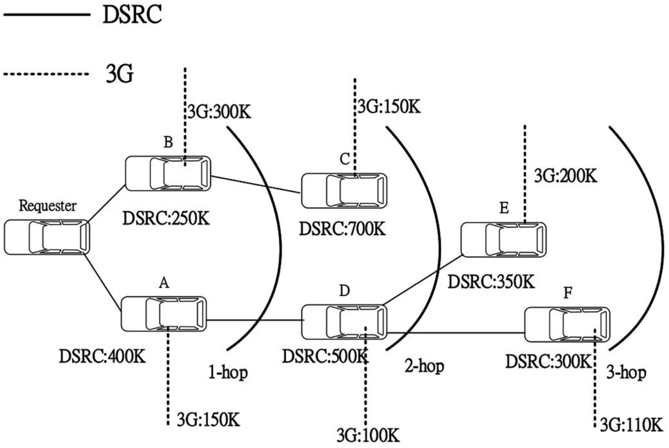


Fig. 2. Example of the GAP.

When selecting helpers from 1-hop members, the requester would firstly select B as its 1-hop helper without verifying any condition and then update the DSRC bandwidth of member B to 0k (DSRC's 250k is totally used). Next, GAP would also select member A as its 1-hop helper and update its DSRC bandwidth to 250k (DSRC's 400k minus 3G/3.5G's150k). Since all members in the 1-hop group have been selected, the requester then seeks helpers from the 2-hop group and firstly verifies member C. When verifying member C, GAP would firstly check whether C has a neighbor been selected as a helper previously. Since it has a neighbor B been selected as a helper, GAP then verify the condition whether adding member C to be a helper would exceed any previously selected helpers' DSRC bandwidth or not. Since, the additional consuming bandwidth caused by member C would exceed helper B's DSRC bandwidth, GAP would skip member C and verify the next ranking member D. Since member D has a neighbor A been selected as a helper and selecting D to be a helper wouldn't exceed any previously selected helpers' DSRC bandwidth, GAP then selects D as a helper and updates A's DSRC bandwidth to 150k (DSRC's 250k minus 3G/3.5G's 100k of D) and D's DSRC bandwidth to 400k (DSRC's 500k minus 3G/3.5G's 100k). Thereafter, GAP starts to select helpers from 3-hop members. It firstly verifies the condition of the highest ranking member E. Although member E has a neighbor D been selected as a helper, selecting E as a helper would exceed the remaining available DSRC bandwidth of helper A. Thus, member E is skipped and the next ranking member F is verified. Since member F has a neighboring helper D and selecting F wouldn't exceed any previously selected helpers' DSRC bandwidth, GAP selects F as a helper and updates A's DSRC bandwidth to 40k (DSRC's 150k minus 3G/3.5G's 110k of F), D's DSRC bandwidth to 290k (DSRC's 400k minus 3G/3.5G's 110k of F), and F's DSRC bandwidth to 190k (DSRC's 300k minus 3G/3.5G's 110k). Since all members have been verified, GAP stops and eventually selects helpers {B, A, D, F} with the aggregated bandwidth 610k.

On the other hand, how to packetize a layered video stream into multiple units using H.264/SVC was resolved in [23]. According to the H.264/SVC standard, the encoded video bitstream is composed of multiple network abstract layer (NAL) units. In the header of each NAL unit, three tags relative to scalability facilities and decoding dependency are (1) Dependency ID (DID), which indicates spatial scalability, (2) Quality ID (QID), which indicates quality/SNR scalability, and (3) Temporal ID (TID), which indicates temporal scalability. We only consider QID and TID in [23]. General speaking, NAL units with QID=N depend on those ones with QID=N-1. Thus, NAL units with QID=0 is defined as base layer (BL). On the other hand, NAL units with QID > 0 is defined as enhancement layer (EL). Since BL is more important than EL, all NAL units belonging to BL (QID=0) are downloaded directly by the requester. Remaining NAL units belonging to EL (QID>0) are downloaded by helpers and then forwarded to the requester.

We utilize a two-tuple NAL(x,y) to represent the corresponding QID and TID of a NAL unit hereafter, in which x denotes the TID and y denotes the QID. Only NAL units with the same TID have decoding dependency. It is not appropriate to assign NAL units with the same TID to different helpers. For example, NAL(1,1) is assigned to a 4-hop helper and NAL(1,2) is assigned to another 1-hop helper. If only NAL(1,2) is received by the requester successfully, it becomes useless and is unable to be decoded. Thus, NAL units with the same TID should be packetized into one assignment unit (AU) that is defined as the basic transmission unit in this paper. Based on the aforementioned description, AU0 is composed of NAL(0,1) and NAL(0,2). AU1 is composed of NAL(1,1) and NAL(1,2). Hence, NAL units belonging to EL can be packetized into one AU array in sequence.

4 STREAMING TASK ASSIGNMENT SCHEME

In the cooperative video streaming scenario, helpers are selected using the greedy approach proposed in [22]. After the phase of helper selection, the requester has chosen some members to be helpers and asked them to provide their redundant 3G/3.5G bandwidth for downloading required video data. However, before asking helpers to download video data, the requester has to decide (1) how to partition the video into multiple streaming tasks for helpers to download and forward video data hop-by-hop to the requester and (2) how to decide appropriate workload for each helper because different helpers are characterized with different capabilities, e.g., different 3G/3.5G bandwidth and hop counts.

At the beginning, the requester will refer the profile of the required video data to package all video data into multiple assignment units. Each assignment unit is the basic unit of a streaming task. In other words, the assigned streaming task of a helper is composed of multiple assignment units. We named the Assignment Unit as an AU which is proposed in [23]. After partitioning the required video data into multiple AUs, the requester then can start the assigning procedure of streaming tasks. First, the requester will do the initial scheduling and determine an assignment interval. The assignment interval is used to decide when the next streaming tasks assignment process is triggered again to assign streaming tasks to helpers. Next, the requester will send requests of initially required video data to helpers through 3G/3.5G network and wait for helpers' forwarding video data during the assignment interval. When the timer of the assignment interval expires, the requester then estimates the appropriate amount of AUs for each helper as the workload of the streaming task for a helper in the next assignment interval. Thereafter, the requester follows a procedure to assign AUs to each helper till reaching the helper's estimated capable amount of AUs during the next assignment interval. After determining the streaming task for each helper, the requester sends the request to each helper again through 3G/3.5G networks and waits for the assignment interval to repeat the estimation procedure again. This procedure will be repeated until all AUs of the video data have been assigned.

In the remaining part of this section, we will discuss how the requester determines the assignment interval used to trigger the assigning process repeatedly. Finally, we explain the procedure used to assign streaming tasks to helpers.

4.1 Assignment Interval

The assignment interval is used to assign streaming tasks to helpers periodically. Before actually assigning streaming tasks to helpers, the requester has to determine the assignment interval to repeatedly trigger the procedure of assigning streaming tasks to helpers. However, the duration of this interval should be deliberately designed because a too long assignment interval would cause helpers idle for long time in the interval; on the other hand, a too short interval would cause helpers cannot finish its assigned streaming task during the assignment interval. Therefore, in order to obtain a suitable assignment interval, we try to calculate an appropriate assignment interval for the streaming tasks assignment scheme.

The idea behind an appropriate assignment interval is that this interval should be long enough for all assigned streaming video data of helpers to be transmitted to the requester in time. In other words, before a new assignment round starts, all requested video data in the previous round should be transmitted by helpers. We have the requester to estimate the appropriate assignment interval as follows. At the beginning, we assume that the requester will start video playback when the buffered video data can playback 10 seconds, and those 10 seconds' buffered video data are assigned to 1-hop helpers only. Let the estimated spending time of 1-hop helpers transmitting these 10 seconds' video data is represented as $T_{initial}$. We can approximately estimate $T_{initial}$ in Equation (3), where $Size_i$, $BW^i_{3G/3.5G}$, and BW^i_{DSRC} stand for the data size of initially downloaded streaming task of helper i, 3G/3.5G bandwidth of helper i, and the DSRC bandwidth of helper i, respectively.

$$T_i = max\left(\frac{Size_i}{BW^i_{DSRC}}, \frac{Size_i}{BW^i_{3G/3.5G}}\right), \forall i \in \text{1-hop helpers} \quad (1)$$

$$T_{initial} = max(T_1, T_2, \ldots, T_i) \quad (2)$$
$$, \forall i \in \text{1-hop helpers.}$$

Since the processes of transmitting video data from the server using 3G/3.5G and forwarding by helpers using

DSRC networks are concurrent, we estimate the maximum transmitted time between downloading and forwarding time of each helper. Then, we let the maximum time of all helpers transmitting video data from the server to the requester as $T_{initial}$. However, during 1-hop helpers are transmitting these initial 10 seconds video data, k-hop's helpers (k > 1) can also use their 3G/3.5G resource to download video data. Therefore, after getting $T_{initial}$, the total throughput of k-hop's helpers (k>1), Th_H , can be calculated as Equation (3).

$$TH_H = \sum_i BW^i_{3G/3.5G} * T_{initial}, \forall i \notin \text{1-hop helpers}. \quad (3)$$

In Equation (3), $BW^i_{3G/3.5G}$ stands for the 3G/3.5G bandwidth of helper i. Here, how to assign initial tasks to helpers by the requester is left to be described in the next subsection. As a result, the assignment interval is obtained by estimating the spending time of transmitting all the required amount of video data at the initial round.

When 1-hop helpers are sending initial 10 seconds' video data, K-hop's helpers (K > 1) are also forwarding their downloaded parts of video data to 1-hop helpers and these video data would be buffered at 1-hop helpers until the initial 10 seconds' video data have been transmitted. Hence, we have the approximate duration for transmitting video data downloaded by K-hop helpers (K > 1) as shown in Equation (4).

$$T_{extra} = \frac{TH_H}{\sum_{\forall i \in \text{1-hop helpers}} BW^i_{DSRC}}. \quad (4)$$

Thus, the approximate assignment interval TAI can be calculated in Equation (5).

$$T_{AI} = T_{initial} + T_{extra}. \quad (5)$$

After selecting helpers from members using the greedy approach proposed in [22] and having determined the assignment interval to periodically assign streaming tasks to these selected helpers, this determined assignment interval won't be changed by the requester until the requester uses the greedy approach to re-select helpers. In case the original selected helpers cannot support the required bandwidth of the requester anymore, the requester will use the greedy approach to select another new set of helpers and determine an assignment interval for these helpers again.

4.2 Streaming Tasks Scheduling

In this sub-section, we describe the procedure of assigning streaming task to each helper in detail. As mentioned before, initially, the requester will start to playback until the initial buffered data is received. Here, the initial buffered data is set to be 10 seconds. To begin the streaming task assignment, the requester will firstly have the initial scheduling and decide the assignment interval. At the initial scheduling stage, the requester will let 1-hop helpers be in charge of initial 10 seconds video data and assign AUs using the round-robin method to each 1-hop helper. For streaming tasks to K-hop helpers (K > 1) at the initial scheduling stage, the requester will firstly estimate the appropriate amount of AUs, which is denoted as Num_{AU}, to each K-hop helper (K > 1) using Equation (6), where $Size_{avg\_AU}$ stands for the average size of the AU.

$$Num_{AU} = \frac{BW^i_{3G/3.5G} * T_{initial}}{Size_{avg\_AU}}. \quad (6)$$

After having the estimated amounts of AUs, the requester then assigns tasks hop-by-hop to each helper from 2-hop to K-hop (K > 2). This assigning process will be repeated till all helpers with the largest hop-count have been assigned AUs according to their estimated capabilities. Since all helpers have been assigned streaming tasks, the requester sends requests to helpers through 3G/3.5G and waits for the assignment interval. Until the timer of the assignment interval is expired, the requester then monitors the amount of successfully transmitted AUs from each helper to estimate an appropriately assigned amount of AUs for each helper in the next assignment interval. Let the assigned number of AUs for helper i in the k-th interval be $Est\_AU^k_i$ and the actual received number of AUs downloaded by helper i in the k-th interval be $Recv\_AU^k_i$. The requester will record the number of successful transmission times STT, which indicates the times that a helper has successfully transmitted all its assigned AUs, of each helper. If helper i has successfully transmitted all assigned AUs to the requester in the k-th interval, the requester will increase STT_i by one and estimate the appropriate amount of assigned number of AUs for this helper in the (k+1)-th interval using the following equation.

$$Est\_AU^{k+1}_i = Est\_AU^k_i + 2^{STT_i}. \quad (7)$$

However, if the received number of AUs downloaded by helper i in the k-th interval is smaller than the requested amount AUs to this helper, the requester will ask this helper to be in charge of the number of AUs as the received one in the (k+1)-th interval and set STT_i of helper i to be zero. Thus, Equation (7) can be reduced to Equation (8).

$$Est\_AU^{k+1}_i = Recv\_AU^k_i. \quad (8)$$

Following this rule, the requester now has estimated suitable workload for each helper during the next assignment interval. Just like the procedure of assigning tasks to K-hop helpers in the initial round, the requester then assigns AUs to helpers from 1-hop to K-hop according to the estimated suitable workload for each helper. After determining the actual task for each helper, the requester sends requests to each helper and waits for the assignment interval again. This process will be repeated till the end of the AU array.

As for the streaming task assignment scheme for the requester itself, the requester also periodically schedules streaming task for itself, but the interval used to repeatedly schedule the streaming task is not the assignment interval used for assigning tasks to helpers. In the initial round, the requester will ask initial 10 seconds' base layer data

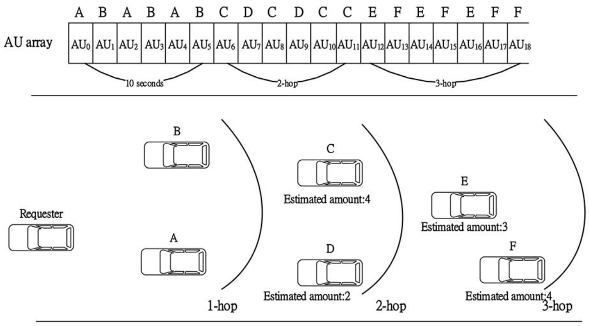


Fig. 3. Example for assigning streaming tasks.

from the video server[2]. After receiving the initial 10 seconds' video data, the requester then starts to playback the video and repeatedly asks video data from the video server periodically. The default time period can be determined by experiments. For example, we estimate the default time period is 10 seconds based on our experiments. Each time the requested number of NAL units from the video server is determined by Equation (9), where Num_{NAL_req} stands for the requested number of NAL units, BR is the average playback bitrate of base layer data, DD is the default time period and $Size_{avg\_NAL}$ stands for the average size of the NAL. If the arriving rate of requested NAL units is always slower than the consuming rate of NAL units, the requester may have to stop playback the video for a while and start to playback again until it has buffered enough data again.

$$Num_{NAL\_req} = \frac{BR * DD}{Size_{avg\_NAL}}. \tag{9}$$

Next, we illustrate the scheduling of the streaming tasks by giving an example of assigning tasks in the initial scheduling stage. Referring to Fig. 3, there are 6 helpers, i.e., A, B, C, D, E, F, that can help the requester to transmit the video data and they are distributed among different hop-count distances. At the initial scheduling stage, the requester will have 1-hop helpers, i.e., A and B, to take the responsibility of downloading initial 10 seconds' video data and then have helpers other than 1-hop helpers, i.e., C, D, E, and F, to be in charge of downloading video data whose presentation time is 10 seconds after. Referring to Fig. 3, initial 10 seconds' AUs are assigned to 1-hop helpers A and B by round-robin. For helpers C, D, E and F, the requester will firstly estimate the appropriate number of AUs for each one. Here, the estimated amount of AUs for C is 4, 2 AUs for helper D, 3 AUs for helper E and 4 AUs for helper F. After determining a suitable amount of workload for each helper, the requester starts to assign tasks to helpers from 2-hop to 3-hop. When assigning tasks to helpers in the 2nd hop, i.e., C and D, the requester uses the round-robin method to one-by-one assign AUs to each helper till reaching its estimated amount of AUs. Referring to Fig. 3, AUs after the initial 10 seconds are alternately assigned to helpers C and D. When the assigned number of AUs for helper D reaches 2 AUs, the requester stops assigning AUs to D but keeps assigning AUs to helper C till its assigned AUs reaches 4 AUs. After helpers in the 2nd hop have all been assigned tasks, the requester assigns AUs to helpers in the 3rd hop. Then, the requester assigns AUs to E and F. After all of these 6 helpers have been assigned streaming tasks, the requester sends the request to each of them to notify them to transmit scheduled video data.

In Fig. 3, the requester can be aware of the configuration of group members by periodically broadcasting a configuration message. After receiving a configuration message, a member uses its carried information to maintain (1) the members own hop-count information, (2) the list of surrounding members and their hop-count distances, and (3) the timestamp of the received configuration message. During the exchanging of configuration messages among members, a member would firstly verify whether the received hop-count distance is usable or not. If it is not, the member just drops this message; otherwise, the member updates its maintained hop-count distance and rebroadcasts a configuration message to notify its neighbors about its updated information.

When member X receives a configuration message sent from member Y, X would firstly check the attached timestamp, which indicates the validity of this configuration message. If the attached timestamp of the received configuration message is smaller (earlier) than the one currently maintained by member X, X adds member Y, which sent this configuration message, into X's surrounding members list and drops the configuration message because this configuration message is not the latest. If the attached timestamp of the received configuration message is bigger (later) than the one currently maintained by member X, member X just updates its maintained information and broadcasts a configuration message carrying the corresponding updated information. If the attached timestamp of the received configuration message is the same as the one currently maintained by member X, member X will further check if it needs to update its maintained hop-count distance. If the carried information of the received configuration message indicates that X's current hop-count distance is smaller than the maintained one if X's packets are passed through Y to reach the requester, member X then updates its hop-count distance to the smaller one and further broadcasts a configuration message carrying the new information to other members. On the other hand, if the carried information indicates that member X has a larger hop-count distance if X's packets are passed through Y to reach the requester, which means that this configuration message has travelled through a longer path, then member X will ignore this configuration message and drop it.

5 PACKET FORWARDING SCHEME

After the requester determining and assigning streaming tasks to helpers, each helper would try its best to transmit the assigned video data to the requester. When video data are forwarding from a K-th hop helper to the requester, all

---

[2] The "x" in the initial x seconds' base layer data, which is 10 in our work, can be set by the system. The bigger/smaller x is set, the longer/shorter initial delay the users will experience and the bigger/smaller jitter the system can tolerate. Thus, the set of x is a compromise of the initial delay and the maximum jitter concerns. That is, depending on the initial delay and the maximum jitter that the corresponding application wants to have, the system can set x accordingly.

of these video data would be received and transmitted hop-by-hop by forwarders. Since these video data are processed by multiple forwarders, the end-to-end transmission quality is greatly affected by each forwarder. In other words, how a forwarder uses its DSRC wireless channel to transmit its currently buffered data to the requester has impact on the streaming video quality.

To observe a forwarder's buffered data, each forwarder may simultaneously receive video data from the video server (in case it's a helper) and other forwarders. A forwarder with the smaller hop-count distance to the requester will buffer more video data than forwarders with the larger hop-count distance because all assigned video data are forwarded to the requester and will be aggregated to forwarders that are close to the requester. Since the DSRC bandwidth resource is limited, forwarders in the smaller hop-count distance are likely to have the data congestion situation. When data congestion happens, data are suffering more waiting time to be sent to the requester and this will indeed affect the streaming video quality. On the other hand, vehicles are mobile and the connection lifetime between two vehicles is not static, the transmission order of buffered data for a forwarder during the limited connection lifetime would also affect the end-to-end transmission time of data and the streaming video quality. As a result, the transmission strategy that a forwarder is used to send its buffered data is important and has to be taken into consideration.

In this section, we discuss the transmission strategy. The transmission strategy is used to manage the usage of the forwarder's DSRC resource for buffered video data of streaming tasks. Here, during the hop-by-hop forwarding procedure of video data, the next forwarder is decided as follows. When a forwarder tries to forward a packet, it randomly picks one helper/member with the smaller hop-count distance from its neighbors and intends to select helpers as the next forwarder first. In our research, how to select an optimized routing path is out of consideration and we concentrates on how to use a forwarder's limited DSRC wireless resource effectively. Since a forwarder may buffer other helpers' downloaded video data and its DSRC bandwidth is limited, this forwarder has to effectively manage the usage of its DSRC resource to let part of buffered video data have higher priority to be forwarded through the DSRC network. In the remaining part of this section, we propose three transmission strategies for a forwarder to manage the usage of its DSRC resource and these three strategies are First In First Out (FIFO) strategy, Playback Priority First (PPF) strategy, and Bandwidth AWare (BAW) strategy.

5.1 First in First Out (FIFO) Strategy

The FIFO is an intuitive transmission strategy which doesn't adopt any consideration to enhance the efficiency of the usage of a forwarder's DSRC resource. In the FIFO transmission strategy, when a forwarder receives a packet, it just inputs this video data into the tail of its buffer. When the forwarder wants to send video data, it always fetches the first one of the buffer to send. However, when a forwarder has received video data of a new round streaming task from the video server or other forwarders, it would drop all left data belonging to the previous round in its buffer because it's time for sending newly assigned video data. The processing procedure of the FIFO strategy is depicted in Fig. 4.

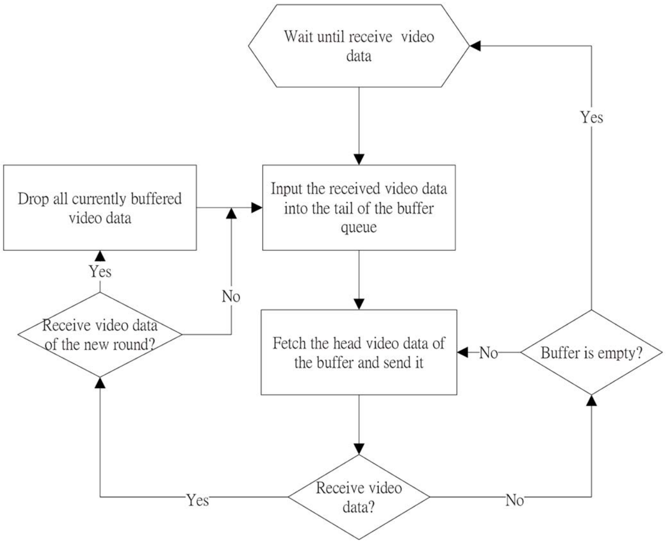


Fig. 4. Processing procedure of the FIFO strategy.

5.2 Playback Priority First (PPF) Strategy

The basic idea of the playback priority first (PPF) transmission strategy is that whenever a forwarder has buffered data that need to be forwarded, it always lets video data having the highest playback priority to use its DSRC resource first. The playback priority stands for the playback order of the video data and is decided when the requester assigns tasks to each helper. As mention before, the requester uses the round-robin method to assign AUs one-by-one to helpers with the same hop-count distance and assigns tasks from 1-hop helpers to K-hop ($K > 1$) helpers. Since AUs are fetched one-by-one from the AU array and are assigned to helpers hop-by-hop, we let streaming tasks of helpers with the same hop-count distance have the same playback priority and NAL units are specified with the same playback priority of the streaming task they belongs to. The playback priority is increased by one as the assigning procedure switches to assign tasks to the next hop helpers. In other words, streaming tasks downloaded by helpers with the same hop-count distance have the same playback priority and their playback priority is higher than streaming tasks of helpers with the larger hop-count distance.

In the PPF transmission strategy, a forwarder will prepare a queue for each streaming task that it has received. Whenever a forwarder receives video data, it will categorize it according to which helper is responsible to download this video data through its 3G network and then inputs this video data into the corresponding queue. In case a forwarder has buffered data of multiple streaming tasks and thus has multiple queues, this forwarder will follow a determination rule to permit video data within particular queues have the right to access its DSRC resource. For the determination rule in PPF, forwarders try to let video data with earlier playback time use the DSRC resource first.

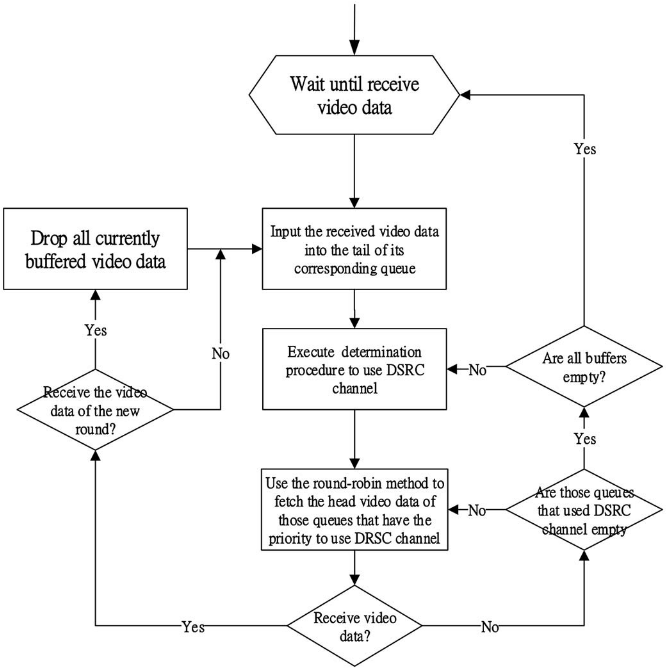


Fig. 5. Processing procedure of the PPF strategy.

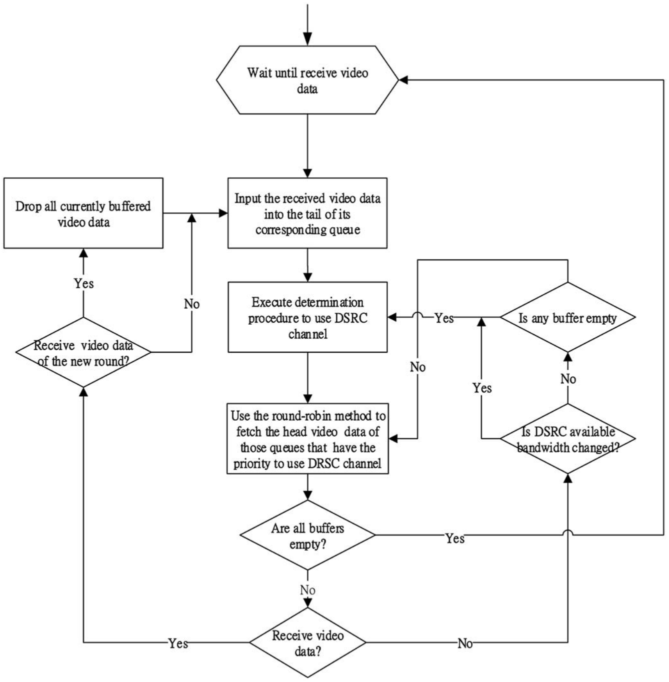


Fig. 6. Processing procedure of the BAW strategy.

As a result, the determination rule of PPF will always let streaming tasks with the highest playback priority have the right to use DSRC resource. The use right of those selected queues will last till all buffered data in those queues are transmitted. Thereafter, this forwarder will follow the determination rule again to permit another set of queues to access DSRC resource. After determining a set of queues which currently have the right to use DSRC resource, whenever the forwarder tries to forward video data, it would use the round-robin method to fetch video data from those permitted queues and send them. The processing procedure of the PPF strategy is depicted in Fig. 5.

5.3 Bandwidth Aware (BAW) Strategy

The basic idea of the bandwidth aware (BAW) transmission strategy is that a forwarder should forward streaming tasks based on their bandwidth requirements. From the aforementioned description, the requester uses the round-robin method to assign AUs one-by-one to helpers with the same hop-count distance and assigns tasks from 1-hop helpers to K-hop ($K > 1$) helpers. Since each forwarder not only has streaming tasks from his 3G/3.5G network but also has multiple streaming tasks from DSRC networks, the forwarder may have streaming tasks with different required bandwidths to be forwarded. Thus, the forwarder needs to maximize the total number of streaming tasks in the proposed BAW strategy.

Similar to PPF, a forwarder will prepare a queue for each streaming task and follow a determination rule to use the DSRC channel in the bandwidth aware (BAW) transmission strategy. However, the determination rule in BAW is different from that of PPF. The proposed BAW strategy is able to let as many streaming tasks to use DSRC resource as possible but to constrain the amount of permitted queues by currently available DSRC resource. In order to let the requester's playback more smoothly, the required bandwidth of incoming video data needs to meet the consuming rate of the video data and hence we use this to indicate the required bandwidth of a streaming task. The required bandwidth of each streaming task can be obtained when the requester has determined the streaming task for each helper. According to this basic idea, we have to recognize the required bandwidth of each streaming task. In order to realize the required bandwidth of each streaming task BW_{req} in the BAW strategy, the consuming rate of a streaming task is considered in Equation (10), where $Size^{i}_{task}$ stands for the total size of this assigned task and PPD represents the corresponding playback period of this streaming task.

$$BW_{req} = \frac{Size^{i}_{task}}{PPD}. \tag{10}$$

For the determination rule in BAW, a forwarder would orderly permit using DSRC channel to forward streaming tasks from lower required bandwidth to higher required bandwidth till filling out currently available bandwidth of DSRC. After determining a set of queues that currently have the right to use DSRC resource, whenever the forwarder tries to forwarder video data, it would use the round-robin method to fetch video data from those permitted queues. The processing procedure of the BAW strategy is depicted in Fig. 6.

6 EXPERIMENT RESULTS

In order to verify our works, the NS2 network simulation tool is adopted to evaluate the performance of the proposed protocol. In the simulation, each mobile node is equipped with two wireless interfaces, in which one is 3G/3.5G cellular interface and the other one is DSRC interface. The 3G/3.5G cellar interface is used to communicate with a

TABLE 1
Parameters Used to Simulate the IEEE 802.11p Interface

Packet Payload	8000 bits	Mac+Phy headers	384 bits
ACK	112 bits	RTS	160 bits
CTS	112 bits	SIFS	32 us
DIFS	58 us	Slot Time	13 us
Propagation Delay	1 us	Channel Bit Rate	6 Mbps (BPSK)

base station that provides the accessibility to the Internet. The IEEE 802.11p interface of each mobile node is used for communication between nodes through ad-hoc network and the settings of corresponding parameters are summarized in Table 1. We set related parameters of IEEE 802.11p into the 802.11Ext module and the WirelessPhyExt module supported in the ns2 and the radio propagation model for the interface is set to be the two-ray ground reflection model.

To simulate our proposed CVS protocol, a YUV video file is encoded into one base layer and three enhancement layers using H.264/SVC. Thereafter, we extract the encoded bitstream to a trace file that records the corresponding information of each extracted NAL unit, and then the trace file is fed to the NS2 to simulate the CVS application. Finally, we have the simulation results through analyzing a trace file that records related information. In the NS2 simulation environment, a highway scenario is constructed to simulate the CVS protocol. In this paper, each member is willing to share its 3G/3.5G bandwidth, it is randomly chosen between 50Kbps to 150Kbps. There are some DSRC contending vehicles in the simulation process.

We evaluate the streaming task assignment scheme proposed in our protocol from two aspects. First, we present the influence of the streaming video quality in different durations of the assignment interval. Secondly, we observe the adaptive capability of our streaming task assignment scheme when the simulation environment is changing. In order to observe the impact of the streaming video quality in different durations of the assignment interval, we compare the resulted video quality using the assignment interval proposed in our work with others' results by randomly selected durations of the assignment interval. In order to observe the adaptive capability of our proposed streaming task assignment scheme, we change the number of DSRC contending vehicles during the video streaming process. We present the adaptive capability of our streaming task assignment scheme by comparing the total assigned size of streaming tasks to the best effort throughput within each assignment round. To get the best effort throughput, we attach each selected helper a traffic generator that generates traffic with a specific rate and forces the helper to have a packet to be transmitted all the time. Therefore, the best effort throughput is obtained by monitoring the amount of received packets at the requester.

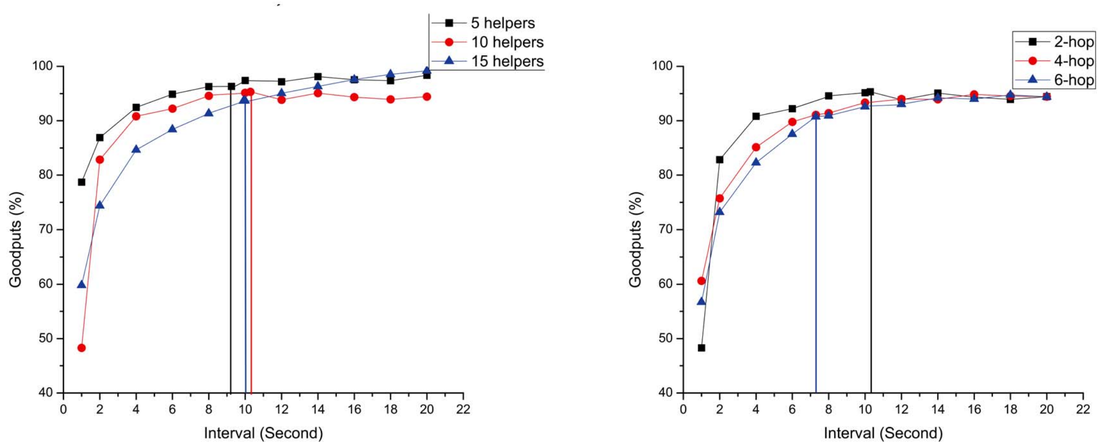
We compare streaming video qualities using three transmission strategies. We also evaluate these three transmission strategies from two aspects. For one aspect, we evaluate those three transmission strategies in simulation environments with different numbers of DSRC contending vehicles. In the simulation, we gradually add contending nodes into the simulation environment to evaluate the performance. For the other aspect, we evaluate those transmission strategies by simulating the streaming qualities under different values of Largest Hop-Count Distance between a helper and the requester (LHCD). In this simulation, we let the number of selected helpers be 10 and extend the largest hop-count distance between the requester and selected helpers from 2-hop to 10-hop.

Among these simulations, we consider the goodput in our experiments. The goodput is the proportion of the decodable number of received NAL units to the total assigned NAL units of helpers. Let Num_{dec_NAL} stand for the decodable number of received NAL units and Num_{total_NAL} stand for the total assigned NAL units. Then,

$$goodput = \frac{Num_{dec\_NAL}}{Num_{total\_NAL}}. \tag{11}$$

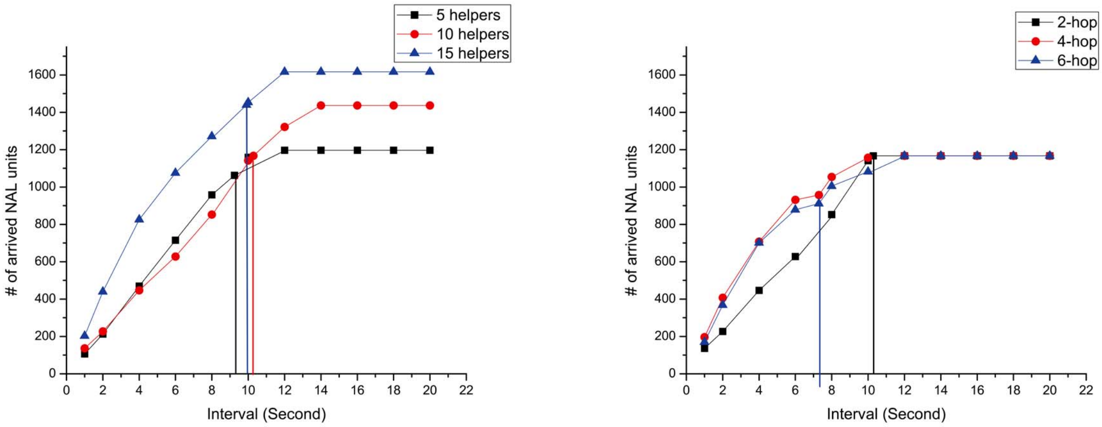
Fig. 7(a) and (b) show the goodputs of different interval among different hops and helpers. In Fig. 7(a) and (b), vertical lines indicate the calculated assignment intervals proposed in our work and each forwarder adopts the FIFO transmission strategy to forward packets. From Fig. 7(a) and (b), we can see that the duration of the assignment interval does have impact to the video streaming quality. When the assignment interval is set to be 1 second, the obtained streaming quality is generally bad. It's because the duration of the assignment interval is too short to transmit assigned streaming tasks back in time. Besides, since the amount of NAL units that can be transmitted back by helpers is limited by the short interval, the total amount of NAL units that can be assigned by the requester is small. Hence, NAL units cannot be transmitted back before the decoding deadline. As the duration is getting longer, the streaming quality is getting improved. However, the improved degree is decreasing as the duration is getting longer and finally the obtained goodput will be limited at a bound. The reason for this phenomenon is that (1) as the duration of the assignment interval is getting longer, assigned streaming tasks are almost transmitted to the requester in time and (2) enlarging the duration of the interval cannot get obvious improvement anymore. We present the amount of received NAL units within the initial round in Fig. 8(a) and (b). From Fig. 8(a) and (b), we can see that as the duration of the assignment interval is getting larger, the number of received NAL units at the requester is getting larger but is constrained at the total assigned amount of NAL units within the initial round. However, when the duration is long enough to transmit assigned streaming tasks, it's unreasonable to enlarge the duration of the assignment interval because it will idle the aggregated resource from helpers. Observing those calculated assignment intervals in Fig. 8(a) and (b), we find that our estimated assignment interval is reasonable enough to transmit assigned NAL units and those calculated values aren't unreasonably large.

In Fig. 9(a) and (b), we gradually add the number of DSRC contending vehicles into the simulation environment to compare these three transmission strategies under different numbers of DSRC contending vehicles. In Fig. 9(a)



(a) (b)

Fig. 7. Goodputs of different intervals. (a) Different helpers. (b) Different hops.



(a) (b)

Fig. 8. Numbers of arrived NAL units in the initial assignment round. (a) Different helpers. (b) Different hops.

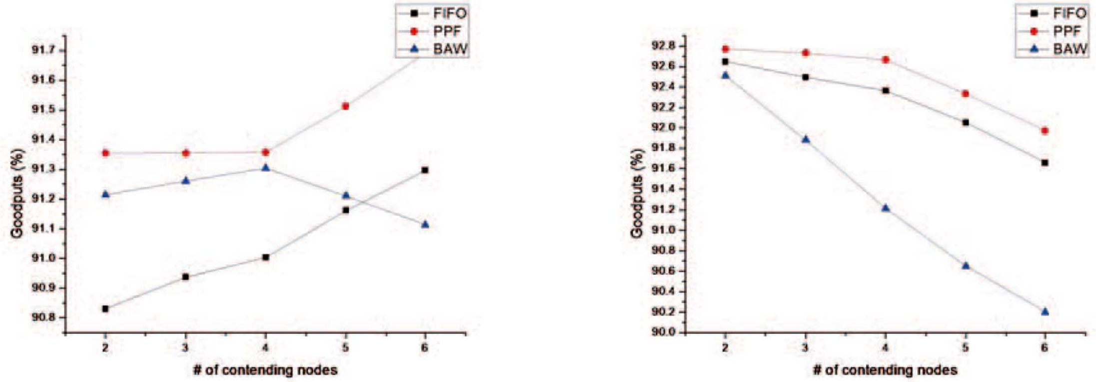


Fig. 9. Goodputs of different numbers of contending nodes among three methods. (a) Large hop count distance between a helper and the requester = 4. (b) Large hop count distance between a helper and the requester = 8.

and (b), the large hops count distance between a helper and the requester are 4 and 8. These results are obtained from a few times' simulations in average. As we can observe from these two figures, (1) the PPF transmission strategy can generally provide better video streaming qualities than the other two transmission strategies and (2) the FIFO strategy can provide better quality than the one resulted in using the BAW strategy. Since the PPF transmission strategy always tries to let streaming tasks with higher playback priority to be forwarded first, streaming video data are more likely to arrive in order than the other two transmission strategies. Therefore, the PPF transmission strategy can generally have higher number of NAL units, which arrived before decoding deadline, than the other two strategies and hence can have better goodput. Since the BAW transmission strategy doesn't consider the playback order

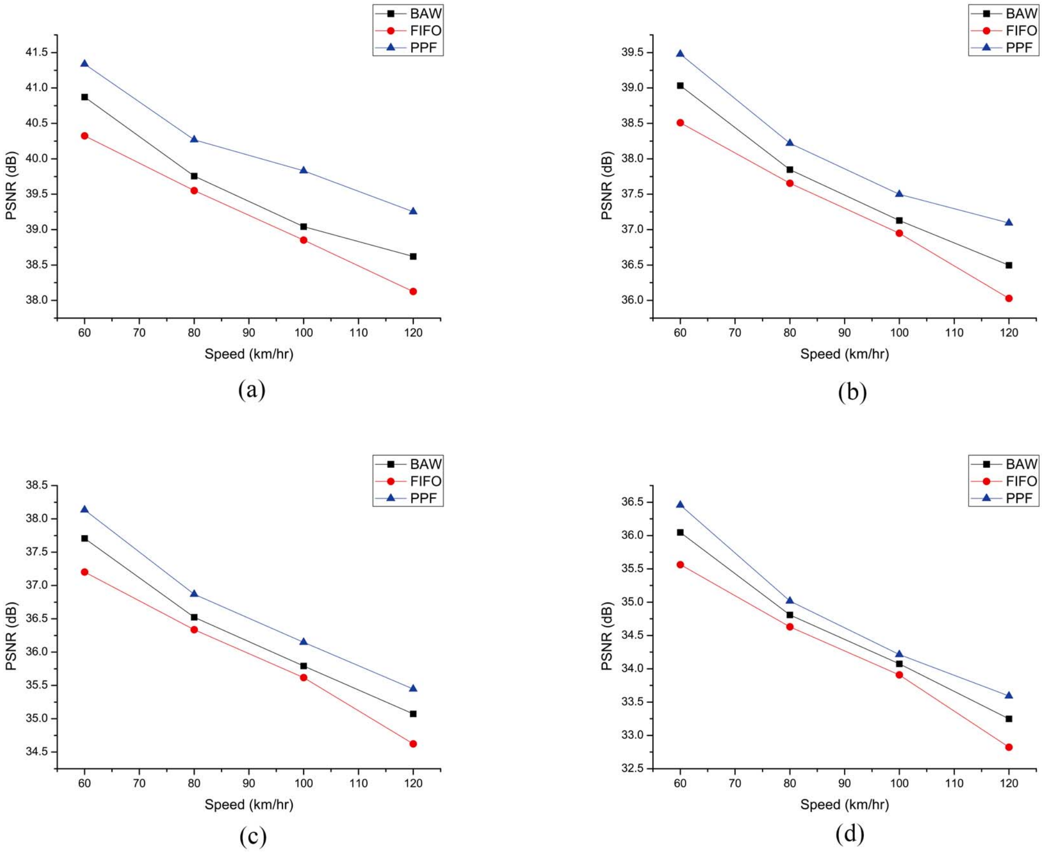


Fig. 10. PSNR comparison in different 3G/3.5G background traffic. (a) 3G/3.5G background traffic: 0 kbps. (b) 3G/3.5G background traffic: 30 kbps. (c) 3G/3.5G background traffic: 60 kbps. (d) 3G/3.5G background traffic: 90 kbps.

when it's forwarding packets, it generally has bad streaming quality. However, we can notice that sometimes the BAW transmission strategy does have better performance than the FIFO strategy. By observing the streaming process at these resulted points, we find out that video data are forwarded more orderly using the BAW strategy. It happens when the determination rule used by the BAW strategy has chosen streaming tasks with higher playback priority to be forwarded first. Besides, the BAW transmission strategy adopts the idea of the video consuming rate at the requester to enhance the streaming quality. It is the reason that sometimes the BAW transmission strategy can have better performance than the other two transmission strategies.

A 1-dimensional highway traffic flow model is established in the simulation. In order to have a much more realistic highway model, vehicles keep a safe following distance with each other for collision avoidance. Regarding safe driving, a vehicle with the higher maximal speed requires to keep a longer safe following distance. Basically, the traffic laws in Taiwan obey the 2-second rule. For example, if the speed of a vehicle is 80 km/hr (about 22 m/s), the suggested safe following distance is 22m/s*2-seconds=44 m. In order to let drivers calculate quickly, the traffic laws in Taiwan define that the value of the safe following distance is half of the vehicle's speed value, i.e., 40, which is 80/2, in the previous example. Thus, we apply this rule into our simulation topology.

This experiment investigates PSNR values of three packet forwarding schemes. Fig. 10 depicts the simulation results in different 3G background traffic. With the increasing background traffic in 3G links, three schemes have lower and lower PSNR values. In addition, when the speed of vehicles increases, the PSNR values of three schemes decrease. Referring to Fig. 10, the PPF scheme keeps better PSNR values than the other two schemes, and FIFO has the worst PSNR values than the other two schemes.

7 CONCLUSION

This paper proposed a k-hop fleet-based cooperative video streaming (CVS) protocol over the hybrid vehicular networks, which is composed of 3G/3.5G cellular network and DSRC ad-hoc network. The proposed k-hop CVS protocol has focused on the issues belonging to the application layer. First, in order to adapt to the time-varying characteristic of the hybrid vehicular networks, the streaming task assignment scheme considers (1) each helper's 3G/3.5G bandwidth and (2) the hop-count distance between each helper and the requester in this work. Second, in order to transmit video data hop by hop smoothly through the DSRC-based ad-hoc network, three different transmission strategies, i.e., first in first out (FIFIO), playback priority first (PPF) and bandwidth aware (BAW), have been proposed and discussed in this work. Finally, we have evaluated the proposed k-hop CVS protocol using NS2 and utilized two parameters, i.e., (1) goodput that represents the successfully received percentage and (2) the number of NAL units that arrived before the decoding deadline, for performance comparison. Based on our simulation results, the estimated assignment interval is large enough to transmit assigned NAL units. In other words, the proposed scheme can avoid helpers being idle or unable to transmit assigned NAL units in time. In addition, the PPF strategy can have better performance than other two strategies. Therefore, the playback priority is the most important factor to decide which buffered video data should be forwarded earlier.

ACKNOWLEDGMENTS

This research is supported by the National Science Council of the Republic of China, Taiwan under the contract number NSC 101-2219-E-006-002.

REFERENCES

[1] *Standard Specification for Telecommunications and Information Exchange Between Roadside and Vehicle Systems – 5 GHz Band Dedicated Short Range Communications (DSRC) Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, ASTM DSR STD E2313-02, 2002.

[2] J. Luo and D. Guo, “Neighbor discovery in wireless ad-hoc networks based on group testing,” in *Proc. 46th Annu. Allerton Conf. Communication, Control, Computing*, Urbana-Champaign, IL, USA, Sep. 2008, pp. 791–797.

[3] H.-W. Tsai, C. Chen, C.-C. Shen, R.-H. Jan, and H.-H. Li, “Maintaining cohesive fleets via swarming with small-world communications,” in *Proc. IEEE VNC*, Tokyo, Japan, Oct. 2009, pp. 1–8.

[4] R. Khalili, D. L. Goeckel, D. Towsley, and A. Swami, “Neighbor discovery with reception status feedback to transmitters,” in *Proc. 29th IEEE Conf. INFOCOM*, San Diego, CA, USA, Mar. 2010, pp. 2375–2383.

[5] T. Taleb *et al.*, “A stable routing protocol to support ITS services in VANET networks,” *IEEE Trans. Veh. Technol.*, vol. 56, no. 6, pp. 3337–3347, Nov. 2007.

[6] T. Okamura, T. Ideguchi, X. Tian, and T. Okuda, “Traffic evaluation of group communication mechanism among vehicles,” in *Proc. 4th ICCIT*, Seoul, South Korea, Nov. 2009, pp. 223–226.

[7] R. Zoican and D. Galatchi, “Analysis and simulation of a predictable routing protocol for VANETs,” in *Proc. 9th ISETC*, Timisoara, Romania, Nov. 2010, pp. 153–156.

[8] A. Nandan, S. Das, G. Pau, M. Gerla, and M. Y. Sanadidi, “Cooperative downloading in vehicular ad-hoc wireless networks,” in *Proc. 2nd Annu. Conf. WONS*, Washington, DC, USA, 2005, pp. 32–41.

[9] Y. Liu and M. Hefeeda, “Video streaming over cooperative wireless networks,” in *Proc. 1st Annu. ACM SIGMM Conf. Multimedia Systems*, Phoenix, AZ, USA, 2010, pp. 99–110.

[10] G. Ananthanarayanan, V. Padmanabhan, C. Thekkath, and L. Ravindranath, “Collaborative downloading for multi-homed wireless devices,” in *Proc. 8th IEEE Workshop HotMobile*, Mar. 2007, pp. 79–84.

[11] M.-F. Leung and S.-H. G. Chan, “Broadcast-based peer-to-peer collaborative video streaming among mobiles,” *IEEE Trans. Broadcast.*, vol. 53, no. 1, pp. 350–361, Mar. 2007.

[12] D. Fan, V. Le, Z. Feng, Z. Hu, and X. Wang, “Adaptive joint session scheduling for multimedia services in heterogeneous wireless networks, in *Proc. 70th IEEE VTC*, Anchorage, AK, USA, Sep. 2009, pp. 1–5.

[13] M. Y. Hsieh, Y. M. Huang, and T. C. Chiang, “Transmission of layered video streaming via multi-path on ad-hoc networks,” *Multimedia Tools Appl.*, vol. 34, no. 2, pp. 155–177, 2007.

[14] M. F. Tsai, N. Chilamkurti, J. H. Park, and C. K. Shieh, “Multi-path transmission control scheme combining bandwidth aggregation and packet scheduling for real-time streaming in multi-path environment,” *Instit. Eng. Technol. Commun.*, vol. 4, no. 8, pp. 937–945, 2010.

[15] K. Rojviboonchai, Y. Fan, Z. Qian, H. Aida, and W. Zhu, “AMTP: A multipath multimedia streaming protocol for mobile ad-hoc networks, in *Proc. IEEE ICC*, 2005, pp. 1246–1250.

[16] M. Lindeberg, S. Kristiansen, T. Plagemann, and V. Goebel, “Challenges and techniques for video streaming over mobile ad-hoc networks,” *Mult. Syst.*, vol. 17, no. 1, pp. 51–82, 2009.

[17] J. Zhao and G. Cao, “VADD: Vehicle-assisted data delivery in vehicular ad-hoc networks,” *IEEE Trans. Veh. Technol.*, vol. 57, no. 3, pp. 1910–1922, May 2008.

[18] I. Leontiadis and C. Mascolo, “GeOpps: Geographical opportunistic routing for vehicular networks,” in *Proc. IEEE Int. Symp. World Wireless, Mobile Multimedia Networks*, Espoo, Finland, 2007, pp. 1–6.

[19] K. Chebrolu and R. Rao, “Bandwidth aggregation for real-time applications in heterogeneous wireless networks,” *IEEE Trans. Mobile Comput.*, vol. 5, no. 4, pp. 388–403, Apr. 2006.

[20] D. Y. Zhu, M. W. Mutka, and Z. W. Cen, “QoS aware wireless bandwidth aggregation (QAWBA) by integrating cellular and ad-hoc networks,” in *Proc. 1st Int. Conf. Quality Service Heterogeneous Wired/Wireless Networks*, Oct. 2004, pp. 156–163.

[21] P. Sharma, S. J. Lee, J. Brassil, and K. Shin, “Handheld routers: Intelligent bandwidth aggregation for mobile collaborative communities,” in *Proc. 1st Int. Conf. Broadband Networks*, Oct. 2004, pp. 537–547.

[22] C.-M. Huang, C.-C. Yang, and H.-Y. Lin, “A K-hop bandwidth aggregation scheme for member-based cooperative transmission over vehicular networks,” in *Proc. 17th IEEE ICPADS*, Tainan, Taiwan, 2011, pp. 436–443.

[23] C.-H. Lee, C.-M. Huang, C.-C. Yang, and H.-Y. Lin, “K-hop packet forwarding schemes for cooperative video streaming over vehicular networks,” in *Proc. 4th Int. Workshop Multimedia Computing Communications-21st ICCCN*, Munich, Germany, 2012, pp. 1–5.

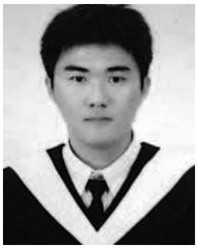
[24] Z. Guan, T. Melodia, and D. Yuan, “Jointly optimal rate control and relay selection for cooperative video streaming in wireless networks,” in *IEEE/ACM Trans. Netw.*, vol. 21, no. 4, pp. 1173–1186, Aug. 2013.

[25] M. Xing and L. Cai, “Adaptive video streaming with inter-vehicle relay for highway VANET scenario,” in *Proc. IEEE ICC*, Ottawa, ON, Canada, 2012, pp. 5168–5172.

Chao-Hsien Lee received the B.S. and Ph.D. degrees with the Department of Computer Science and Information Engineering, National Cheng Kung University, Taiwan, in 2002 and 2007, respectively. He is currently an Assistant Professor in the Department of Electronic Engineering, National Taipei University of Technology, Taiwan. His current research interests include ubiquitous computing, cloud computing, and tele-healthcare embedded system.

Chung-Ming Huang received the B.S. degree in electrical engineering from National Taiwan University, Taiwan, in 1984, and the M.S. and Ph.D. degrees in computer and information science from the Ohio State University, Columbus, OH, USA, in 1988 and 1991. He is currently a Distinguished Professor in the Department of Computer Science and Information Engineering, National Cheng Kung University, Taiwan. He serves as the Director for the Promotion Center for the Network Applications and Services Platform Technology (PC-NASPT), Ministry of Education (MOE), Taiwan. He has published over 300 referred journal and conference papers in wireless and mobile communication protocols, interactive multimedia systems, audio and video streaming and formal modeling of communication protocols. His current research interests include wireless and mobile network protocol design and analysis, media processing and streaming, web technologies, and network applications and services. He is a senior member of IEEE and ACM.

Chia-Ching Yang received the B.S. degree from the Department of Computer Science and Information Engineering, National Cheng Kung University, Taiwan, in 2006. He is currently pursuing the Ph.D. degree in the Department of Computer Science and Information Engineering, National Cheng Kung University, Taiwan. His current research interests include P2P overlay networks, session mobility, and multimedia streaming protocols.



Hsiao-Yu Lin received the B.S. and master's degree from the Department of Computer Science and Information Engineering, National Cheng Kung University, Taiwan, in 2009 and 2011, respectively. His current research interests include vehicular networks and multimedia streaming protocols.