

QoE-driven User-centric VoD Services in Urban Multi-homed P2P-based Vehicular Networks

Changqiao Xu, *Member IEEE*, Futao Zhao, Jianfeng Guan, Hongke Zhang and Gabriel-Miro Muntean, *Member IEEE*

Abstract—Recently many cities around the world have witnessed large scale deployment of terrestrial broadcasting mobile TV to vehicles. This service is similar to the cable or satellite television already watched at home and user-centric interactive mobile Video-on-Demand (VoD) over urban vehicular networks is in fact expected. However providing this new service with focus on user Quality of Experience (QoE) constitutes a significant challenge. This paper introduces a QoE-driven User-centric solution for VoD services in urban vehicular network environments (QUVoD). QUVoD relies on a multi-homed hierarchical peer-to-peer (P2P) and vehicular ad-hoc network (VANET) architecture. Vehicles construct a low layer VANET via WAVE interfaces and also form an upper layer P2P Chord overlay on top of a cellular network via 4G interfaces. A novel grouping-based storage strategy is proposed which distributes uniformly the video segments along the Chord overlay, reducing segment seeking traffic, while also enabling load-balancing. A novel segment seeking and multipath delivery scheme is also introduced which achieves high lookup success rate and very good video data delivery efficiency. Furthermore, a new speculation-based pre-fetching strategy is proposed, which analyses users' interactive viewing behavior and, by estimating video segment playback order, employs pre-fetching of the expected segments, smoothening the video playback. Simulation results show how QUVoD

is a highly efficient user-centric mobile VoD solution in urban vehicular networks in comparison with existing state of the art solutions.

Index Terms—User-centric VoD, peer-to-peer (P2P), quality of experience (QoE), vehicular network.

I. INTRODUCTION

LATELY, many cities around the world have witnessed large scale deployment of terrestrial broadcasting mobile TV services. For example, following the Beijing Olympics, almost all taxis (out of the over 700 thousand vehicles in the Chinese capital city) are equipped with on-board equipment which supports terrestrial TV broadcasting signal retrieval and multimedia playback. In terms of quality, this is like the passive cable or satellite television programs that people watch at home. Additionally we are witnessing extensive developments in wireless access technologies including WiFi, LTE, LTE-A, WiMAX, *etc.* [1] and especially in vehicular wireless technologies such as Wireless Access in the Vehicular Environment (WAVE) (IEEE 802.11p), enabling data delivery via vehicle-to-vehicle (V2V) [2], vehicle-to-infrastructure (V2I) and vehicle-to-Road-Side-Unit (V2R) communications [3]. This paves the way towards vehicular Video-on-Demand (VoD), as mobile VoD is expected to become an attractive service over urban vehicular networks.

Peer-to-peer (P2P) multimedia content sharing from multiple sources to multiple destinations has been shown to be robust and highly scalable, based on distributed solutions in the wired Internet [4]-[12]. Important research efforts are put to extend P2P content delivery and its performance benefits to the wireless domain. Lately P2P-based multimedia services over vehicular ad-hoc networks (VANET) have become a very interesting research topic [13]-[16]. However, although drawing increasing attention from the research community, studies on P2P media delivery over VANETs have been limited to simple sequential remote playback scenarios, which are similar to the broadcast mobile TV, where the passive user viewing pattern simplifies system design. However, user-centric mobile VoD services supporting interactivity can offer edited video clips to passengers on demand, during their trips, extending the existing services offered by the likes of YouTube [4] for example from the office or home to the road. This represents one of the most innovative trends and value-added services in urban vehicular networks.

In this context, providing support for high quality of experience (QoE) for user-centric mobile VoD service in VANETs

Copyright (c) 2012 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

Manuscript received June 22, 2012; revised October 3, 2012 and November 6, 2012; accepted November 12, 2012. This work was supported in part by the National High-Tech Research and Development Program of China (863) under Grant No. 2011AA010701, by the National Basic Research Program of China (973 Program) under Grant 2013CB329100 and 2013CB329102, by the National Natural Science Foundation of China (NSFC) under Grant No. 61001122, 61003283 and 61232017, by the Beijing Natural Science Foundation of China under Grant No. 4102064, by the Jiangsu Natural Science Foundation of China under Grant No. BK2011171, by the Jiangxi Natural Science Foundation of China under Grant No. 20122BAB201042, and by the Fundamental Research Funds for the Central Universities under Grant No. 2012RC0603 and 2011RC0507. The partial support of Lero (grant number 10/CE/I1855) is gratefully acknowledged.

C. Xu is with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China. He is also with the Institute of Sensing Technology and Business, Beijing University of Posts and Telecommunications, Wuxi, Jiangsu, China (e-mail: cqxu@bupt.edu.cn).

F. Zhao and J. Guan are with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China (e-mail: zhaofutao@bupt.edu.cn, jfguan@bupt.edu.cn).

H. Zhang is with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China. He is also with the National Engineering Laboratory for Next Generation Internet Interconnection Devices, Beijing Jiaotong University, Beijing, China (e-mail: hkzhang@bupt.edu.cn).

G.-M. Muntean is with the Performance Engineering Laboratory, School of Electronic Engineering, Network Innovations Centre, Rince Institute, Dublin City University, Dublin, Ireland (e-mail: munteang@eeng.dcu.ie).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

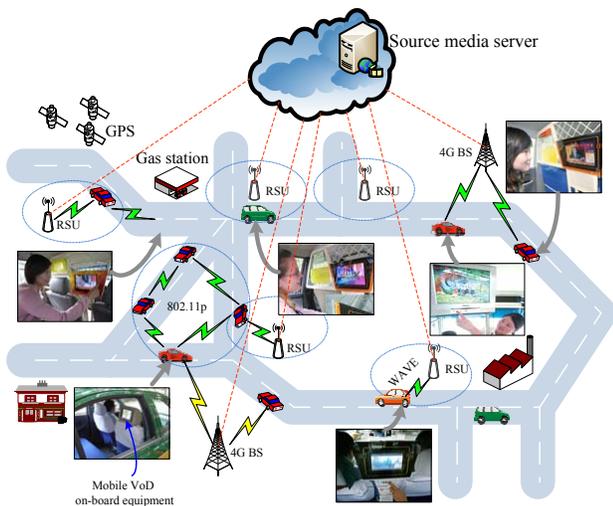


Fig. 1. Quality-oriented User-centric mobile VoD system for vehicular networks.

is a significant challenge. As we know, end-to-end video transmission quality plays an important role in maintaining high levels of user QoE in P2P-based VANETs. Most existing solutions disseminate video data via the underlying vehicular ad-hoc network in multi-hops [13]-[16]. However, VANETs easily become disconnected due to node mobility, leading to unreliable transmissions and negatively affecting user QoE levels. The latest technological advances enable vehicular devices to be equipped with multiple types of wireless interfaces in order to support access to different networks, forming a multi-homed heterogeneous network environment. During communications, the source nodes can select and simultaneously use for data delivery one or several paths to the destination. The selection is performed dynamically following changes in the availability and/or characteristics of the access networks behind nodes' interfaces. This significantly increases fault tolerance and supports provision of services with increased user QoE levels.

On the other hand, supporting user interactive viewing behavior in VoD services requires the proposal of a mechanism to enable smooth user video playback by fetching in advance the video segments in line to be played into the vehicle's buffer. The pre-fetching has become a default strategy and has been widely used in current Internet-based and wireless-based P2P multimedia delivery systems. However most of them only pre-fetch the subsequent video segment for normal playback in strict sequential order. When VCR-like operations occur, searching for the newly requested segment has to happen and once found it has to be downloaded, decoded and then played back. This procedure is associated with long latency and seriously deteriorates users' viewing QoE.

This paper introduces a novel efficient interactive Quality-oriented User-centric mobile VoD solution for vehicular networks (QUVoD), illustrated in Fig. 1. QUVoD relies on a newly designed multi-homed hierarchical P2P/VANET architecture and makes use of highly innovative algorithms for *video segments storage, video resource search, multipath data transfer* and *speculative segment pre-fetch* to address the

challenges of vehicle mobility and users' interactive viewing behavior. These novel algorithms will be described in details in this paper alongside simulation-based performance evaluation. Testing results fully illustrate how QUVoD is an efficient user-centric mobile VoD solution in vehicular networks, which outperform existing state-of-the-art solutions.

II. RELATED WORK

Recently, P2P content sharing for multimedia streaming services in vehicular networks has attracted increasing research interests from various scientists. Hsieh *et al.* [13] proposed an effective dynamic overlay multicast strategy for live multimedia streaming in urban VANETs. Zhou *et al.* [14] developed a P2P media-service scheme that jointly solves the content dissemination, cache update, and fairness problems for vehicular networks. Qadri *et al.* [15] presented a realistically modeled scheme for streaming video over VANETs, by means of an overlay network with multiple sources. Using symbol-level network coding, Yang *et al.* [16] introduced a live multimedia streaming system CodePlay in VANETs. However, none of the above works considered the vehicular interactive VoD service, in which users can jump to any watching point according to their interests. The interactive service significantly enhances user personal viewing experience, but introduces important technical challenges, which need to be addressed.

Interactive P2P Video on Demand (VoD) over Internet is currently gaining momentum in the research community [5]-[12]. We previously proposed BBTU [5] and SDNet [6] for tree-based unstructured P2P networks to address VCR-related issues in wired non-mobile network environments. By adopting a mesh-based topology, Chang *et al.* [7] introduced an interleaved video frame distribution (IVFD) scheme to support full VCR functionality by making use of gossip messages to search for resources. Zhou *et al.* [8] introduced a unifying request scheduling model and content replication strategy to minimize server load. Lee *et al.* [9] proposed a popularity-aware pre-fetching scheme to support interactive P2P Streaming. Wu *et al.* [10] investigated how movie popularity can impact server's workload and optimized the replication strategy of P2P-VoD systems.

Chord is a well known P2P content distribution architecture. Nodes in Chord have embedded a "precursor-successor" relationship. By associating a key with each data item (i.e. video clip or video segment) and storing the key/data item pair at the node to which the key maps, efficient data storage and localization can be implemented [17]. For example, a Chord-based efficient interactive P2P VoD system named VMesh was introduced in [11]-[12]. All above works address some important issues for P2P VoD such as dissemination topology, video resources storage and search, support for VCR-like operations, video content replication, etc. However, so far, the research of interactive P2P VoD solutions lacks mobility support and to the best of the authors' knowledge does not focus at all on vehicular wireless networks.

P2P models can be generally classified in two types: unstructured and structured. Unstructured P2Ps are mainly based on gossip and flood mechanisms, which produce large amounts

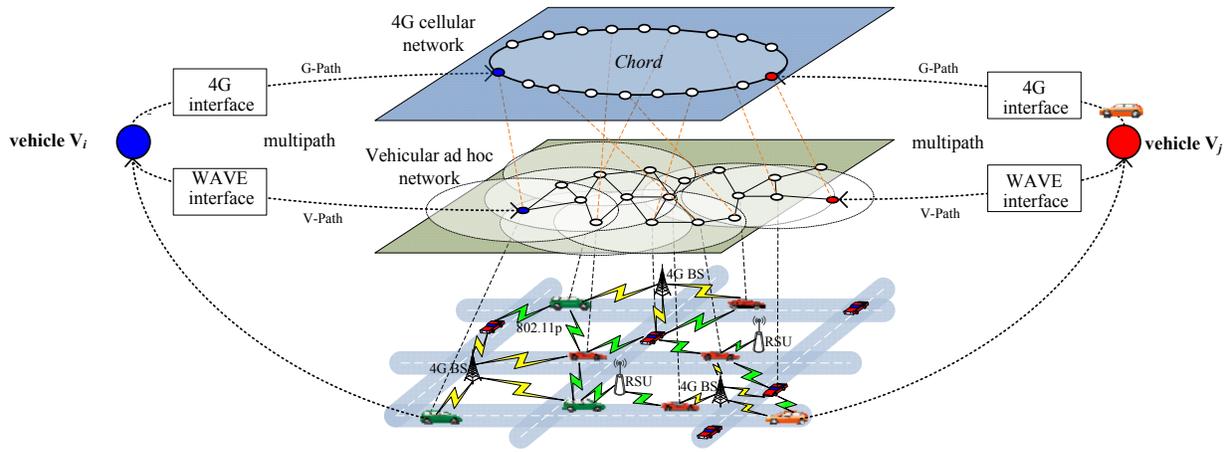


Fig. 2. QUVoD multi-homed hierarchical architecture.

of communication overhead. Most structured solutions use distributed hash tables (DHT) and as the typical lookup length of DHTs is $O(\log(N))$, where N is the number of nodes in the system, the structured P2P solutions have high performance in terms of resource search. MeshChord [18] improves the basic Chord to efficiently utilize it in wireless mesh networks. Liu *et al.* [19]-[20] proposed MChord to enhance the performance of Chord over VANETs. There are also other research works [21] and [22] which improve the Chord algorithm trying to adapt it to the mobile environment. However, all the application-layer P2P Chord networks of the above solutions are built directly on top of the wireless ad-hoc network. It makes Chord seriously unstable as the ad-hoc network links are disconnection prone and this happens frequently caused by nodes' mobility and vehicles' different moving directions.

Recently multi-homed networks have attracted extensive academic research interests. Having multiple network interfaces, end hosts can establish multiple connections with other end hosts across different networks [23], situation which protects the communications from the failure of any one of these networks. M-MIP [24] extended Mobile IP to enable mobile hosts to use multiple care-of addresses simultaneously. Shim6 [25] inserts a sub-layer into the network layer to support network layer multi-homing and mobility. Okimoto *et al.* [26] described functional requirements and possible solutions for multi-homing without the use of NAT in IPv6 for hosts. MPTCP [27] modifies TCP and implements a multipath transport within a transport connection. In [28]-[30], we analyzed and proposed concurrent multipath video transfer solutions by making use of a multi-homing transport protocol SCTP. It can be noted that multipath transport over multi-homed networks increases the efficiency of the resource usage, and thus increases the network capacity available to end hosts.

Based on the latest research progress in P2P multimedia content sharing in VANETs, Internet-based P2P VoD and multi-homed wireless networks, we introduce QUVoD, a novel mobile VoD solution in urban VANETs. To the best of the authors' knowledge, this is the first work to consider providing efficient P2P-based VoD services in urban multi-homed vehicular networks.

III. QUVoD SYSTEM DESIGN

A. QUVoD Architecture

Fig. 2 shows the architecture of QUVoD. We assume vehicles in QUVoD are equipped with dual wireless interfaces: a 4G cellular network interface (e.g. WiMAX, LTE-A) and a WAVE interface (supporting V2V communications). These vehicles construct a multi-homed network, highly feasible to be built in reality. The cellular network is a standard for mobile telecommunications in wide-area mobile environments and supports real-time multimedia services [31]. A cellular network has relatively high stable connectivity merits. However, 4G are normally provided by Internet Service Providers (ISP), which charge fees for data traffic. By making use of the WAVE interface in VANET, vehicles can forward data to their neighboring vehicles or RSUs, and data can transverse intermediate vehicles via ad-hoc communications with low network usage costs. However, VANETs easily become disconnected in situations with low vehicle density and high mobility, (i.e. high speed and/or different driving directions). Taking into account these factors, QUVoD constructs a multi-homed hierarchical P2P/VANET architecture, which employs both Chord and vehicular ad-hoc networks, as shown in Fig. 2. Vehicles construct low layer VANETs via WAVE interfaces for V2V communications. The upper layer Chord overlay is formed on top of the cellular infrastructure via the 4G interfaces. Vehicle V_i can create two connections with vehicle V_j via the 4G path (G-Path) and VANET path (V-Path), respectively. G-Path denotes packet transfer through the 4G infrastructure networks with high stability, but also high cost. V-Path indicates packet delivery over the vehicle ad-hoc network via V2V communications with increased instability, but low cost. QUVoD transmits video data between vehicles V_i and V_j employing a multipath transfer mechanism. As 4G is stable, when vehicle V_i requests a video segment for its future playing, the request message is routed by the DHT for looking up the segment in the Chord network, which means in multi-homed QUVoD, the G-Path is used for resource search. Both G-Path and V-Path are employed for video data multipath transmissions.

The multi-homed hierarchical P2P/VANET architecture de-

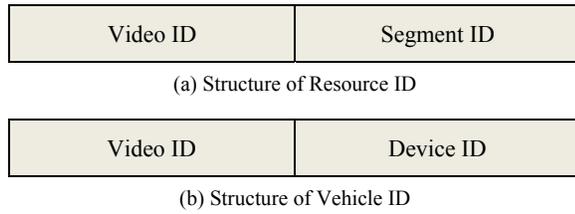


Fig. 3. Structure of Resource ID and Vehicle ID.

sign of QUVoD is highly effective. In most current research, vehicles form an application-layer P2P network on top of the VANET directly and P2P communication relies on the routing protocol of the underlying VANET [13]-[16]. With vehicle movement, the VANET topology changes quickly, which leads to P2P topology changing as well. Consequently both resource lookup and data transfer fail when network disconnections between vehicles occur. This requires frequent redistribution and re-storing of resources between different nodes as the nodes frequently join and leave the system due to their mobility. Unlike these solutions, QUVoD establishes the Chord network communications over the 4G infrastructure network, which makes the P2P topology relatively stable. When a vehicle stores some video segments after it joins the system, it can carry the assigned segments all the time regardless of the driving requirements. For instances, in Beijing, each taxi normally has two drivers who work in shifts, which means the taxi can be 24-hours online. This enables for the target vehicle which stores the required video segments to always be located in Chord through underlying 4G network instead of the failure prone VANET. In other cases, in urban environment, taxi density, traffic congestion and light control in intersection provide video sharing opportunities via V2V VANET communications directly, which can make use of VANETs' own data distribution ability. The multipath transfer mechanism over G-Path and V-Path increases fault tolerance and improves data transfer capabilities. In this context, QUVoD overcomes the drawbacks of each of the 4G and VANET technologies, respectively and makes full use of the advantages of a combined solution.

In addition to the multi-homed structure which enables video segments sharing between vehicles, QUVoD has a media server. This server stores the original video resources and provides service when vehicles cannot fetch video segments from other vehicular nodes. The server also keeps all the video-related information and users can search for the videos in which they are interested. Every video needs to be processed by the server before being distributed over the Chord network. The server assigns each video a unique Video ID according to the HASH algorithm [17]. Besides, for efficiency of management and delivery purposes, it divides each video into several equal size segments, and assigns to each segment consecutive Segment IDs in the order of playback. Consequently, each video segment will have a unique Resource ID, which is composed of the Video ID and the Segment ID as shown in Fig. 3 (a).

Each vehicular node is assigned a unique Device ID by the server using the HASH algorithm when it joins the system.

We define the Vehicle ID as being composed of the Video ID of the video playing and the Device ID of that vehicle as a unique identifier. Fig. 3 (b) shows the composition of Vehicle ID. In QUVoD, all the vehicles playing the same video will have the same Video ID and make up a Chord sub-circle according to the value of vehicular nodes' Device IDs. When a vehicle user is interested in a video, the vehicle node has to join the Chord sub-circle corresponding to the video content sought. This is done by using the existing Chord algorithm [17]. Meanwhile, the vehicle node looks up for the initial video segments, downloads them via the multipath and plays them when the requested data has arrived at its buffer. For video segment search and retrieval, the vehicular node employs the novel *segment seeking and delivery scheme*, which will be described in details in the next section. After joining the system, the node then continues to request and download the next segments by making use of our newly proposed *speculation-based pre-fetching strategy*. When not all the bandwidth is in use, the vehicle will use its available bandwidth to download and store video segment(s) in its local storage using the QUVoD's novel *distributed grouping video segments storage scheme*. The vehicle leaving the system is treated based on the classic Chord leaving algorithm.

B. Distributed Grouping-based Video Segments Storage

In the original Chord design, when the number of nodes is larger than that of video segments, there will be some nodes which do not store any resource, as each video segment is stored in just one node. This creates load imbalance. Furthermore, if a node leaves or breaks down, although the video segment key is still stored in the system, the video segment resource is not accessible. Pitoura *et al.* [32] proposed a fair load distribution solution over DHTs using replication and multi-rotation hashing mechanism. A similar method is employed by VMesh [11]-[12]. In VANET environments, due to vehicular nodes' high dynamicity, it is difficult to guarantee a requester to properly obtain resources from only one supplier. Extending these works [32] and [11]-[12], QUVoD designs a distributed grouping-based storage scheme. It allows any video segment to be stored in multiple nodes and each node to store multiple video segments. This not only enhances P2P network's stability, but also balances nodes' load.

As Fig. 4 shows, in QUVoD, the vehicular nodes in a Chord sub-circle are divided into multiple groups and the number of groups is equal with the number of video segments. The nodes in the same group will store several consecutive segments starting with the same segment. Denote the number of segments as S_{max} , and the size of the Device ID field as L . Equation (1) gives the size of each group L_{group} :

$$L_{group} = 2^L / S_{max} \quad (1)$$

where 2^L is the maximum value of the Device ID and L_{group} is a constant value in a specific Chord sub-circle network because the variables L and S_{max} are stationary. Assuming V_i and D_i are the latest vehicle node that has joined into the network and its ID, respectively, the ID to be assigned to the next node (i.e., V_{i+1}) is defined as:

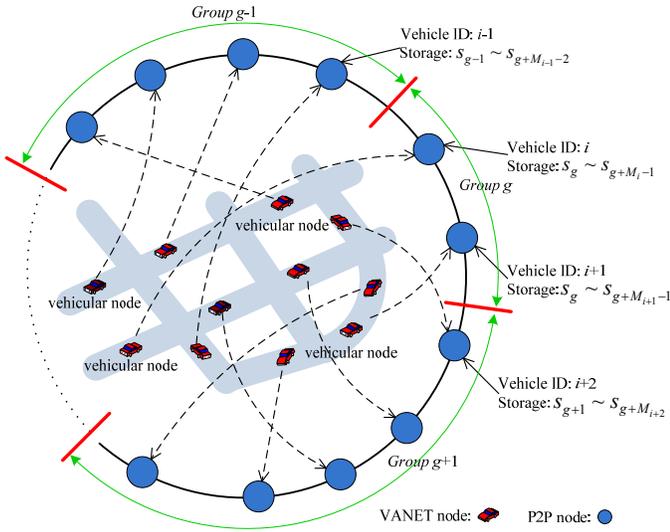


Fig. 4. Grouping-based video segments storage strategy.

$$D_{i+1} = (D_i + L_{group} + 1) \mod 2^L \quad (2)$$

which can have the nodes uniformly distributed over the Chord. We set F_g as the initial boundary Device ID for the g -th group as in equation (3). For instance, the smallest Device ID of the nodes in the first group should be 0. g here also denotes the Segment ID of the starting segment that each node stores in the g -th group.

$$F_g = L_{group} \times (g - 1) \quad (3)$$

We set DS_g as the value set of all possible Device IDs of nodes in group g . DS_g can be defined by equation (4), where N is the set of natural numbers and N^+ is the set of positive integers.

$$DS_g = \{id | id \in N, F_g \leq id < F_{g+1}\} \\ = \{id | id \in N, L_{group} \times (g - 1) \leq id < L_{group} \times g, \\ 1 \leq g \leq S_{max}, g \in N^+\} \quad (4)$$

As already mentioned, QUVoD divides each video in equal size segments, for example $video \Leftrightarrow (s_1, s_2, \dots, s_j, \dots, s_n)$, where j is the Segment ID of segment s_j . Each node in QUVoD stores several successive video segments according to its available memory capacity. The nodes whose Device IDs are in the set DS_g will store the segment s_g and several of s_g 's next segments. Assuming a V_i in group g has a storage memory space M_i , then it could store segment $s_g, s_{g+1}, \dots, s_{g+M_i-1}$, which is indicated in equation (5):

$$VS_i = \{s_j | j \in N^+, \\ g \leq j \leq g + M_i - 1\} \quad (5)$$

where VS_i is the segment set composed of all the segments stored by V_i . Fig. 4 illustrates the distributed grouping-based video segments storage scheme.

When a node still has memory space available after having stored all the segments up to the last segment in the video

Algorithm 1 Video segments storage for vehicle V_i

```

1:  $g = device\_id_i / L_{group}$ ;
2:  $j = g$ ; /*  $j$  denotes the segment ID*/
3:  $idle\_memory = M_i$ ;
   /* $idle\_memory$  denotes  $V_i$ 's remaining memory size*/
4: while ( $idle\_memory \geq sizeof(s_j)$ );
   /* $s_j$  denote a segment whose segment ID is  $j$ */
5:    $V_i$  search and download segment  $s_j$  using algorithm 2;
6:    $j++$ ;
7:   if ( $j > S_{max}$ )
8:      $j = 1$ ;
9:   end if
10:   $idle\_memory -= sizeof(s_j)$ ;
11: end while
    
```

sequence, it starts storing the first segments from the beginning of the sequence until the memory is used up. For example, a vehicle V_i in group S_{max} can store the segment S_{max} and then the segments 1, 2, 3 and so on. Assuming g is the group sequence V_i belongs to, s_j is a segment that V_i will store and $device_id_i$ is V_i 's Device ID, Algorithm 1 illustrates the video segments' storage strategy after V_i joins the system.

C. Video Segments Seek

In QUVoD, a segment is stored in multiple nodes which are distributed in several consecutive groups. For example, the nodes in the g -th group will certainly store the segment s_g , the nodes in the group $g-1$ will have a great probability of storing the segment s_g , the probability of the nodes in group $g-2$ storing segment s_g is smaller than that for the nodes in the group $g-1$, and so on.

In this context we designed a novel segment seeking scheme to obtain segments. According to the QUVoD *video segments storage scheme*, in order to get a segment, we just need to locate a group to which the nodes storing the segment required belong to. We will choose the group in which the nodes have the highest probability of storing the required segment. That is, if we want to get the segment s_g , we will contact the g -th group. We firstly find the first node in the group, and then find other nodes via this first node.

Since all the nodes in the g -th group can serve the segment s_g , a *data scheduling strategy* is designed to balance the load between nodes when serving resources. The first node V_x in g -th group acts as a "scheduler". The "scheduler" maintains a *group member table (GMT)* to record all vehicles' IDs and corresponding serving load information in the g -th group. Any item in *GMT* is associated a 2-tuple $GMT = (NID, Load)$ where *NID* is the Vehicle ID of the member and *Load* is the number of nodes for which this member is data supplier. The "scheduler" is responsible to assign the member with the lowest serving load as supplier to the requester. If V_z joins/leaves the g -th group, V_z 's precursor can be aware of V_z 's variation and reports this change to V_x . V_x will add/remove V_z 's information in *GMT*. The successor V_y of "scheduler" stores and periodically updates a duplicate of *GMT* in order to avoid the loss of *GMT*. If V_x leaves the group, V_y becomes the new "scheduler" and informs the members in the g -th group about this change.

Assuming vehicle V_i is playing a video and requests a segment s_g , it performs the following procedure:

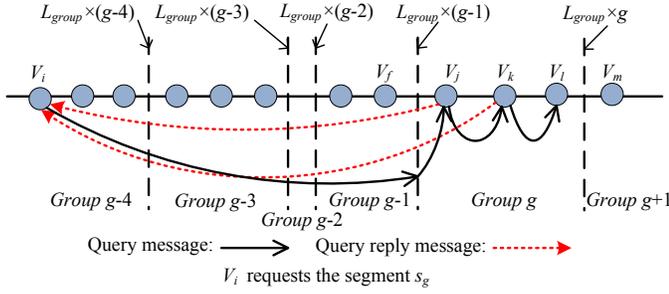


Fig. 5. Video segment seeking procedure.

1) V_i sends out a *Query* message by DHT for locating the node who possesses a group whose initial boundary is F_g by equation (3). According to the Chord algorithm, V_i could find the node whose Device ID equals or is closest but higher than the group's initial boundary F_g along the Chord sub-circle. As the Fig. 5 shows, we can find the first V_j in group g . Denote that, in Fig. 5, we have omitted the Video IDs for Vehicle ID, which are all the same in a Chord sub-circle.

2) When V_j receives the *Query* message, by checking its *GMT*, it selects a member V_k with the lowest load as V_i 's supplier. V_k will reply with a *Query Reply* message to V_i and serve s_g to V_i . V_j modifies V_k 's load value in its *GMT*.

3) If V_i cannot locate the nodes which have the required s_g through the above procedure, it will repeat the process to find nodes in groups $g-1$, $g-2$, ..., $g-M+1$ using initial group boundary F_{g-1} , F_{g-2} , ..., F_{g-M+1} whose values are $L_{group} \times (g-2)$, $L_{group} \times (g-3)$, ..., $L_{group} \times (g-M)$, respectively. The parameter M is the maximum storage limit of all nodes in QUVoD.

Assuming V_k is the located vehicle node after the above search process, the vehicle V_i will download the required video segment s_g from vehicle V_k through G-path and V-Path by taking advantage of the multipath data delivery mechanism which will be discussed in Section III.D. After downloading s_g , V_i also can continue to download s_{g+1} , ..., s_{g+M_k-1} from V_k employing multipath directly, which avoids relocating s_{g+1} , ..., s_{g+M_k-1} in Chord. Algorithm 2 describes the process of video segments seek and download.

To maintain *GMT* and respond to segment requests, the "scheduler" needs to exchange additional messages, which bring additional load. However, the group members storing multiple sequential segments can serve more than one segment in a row. This supports short-range sequential movement of the playback point, which reduces significantly the number of segment request messages sent to "schedulers". In addition, compared with the high-bandwidth streaming traffic, this overhead is light-weighted. Furthermore, we can reduce the load on the "scheduler" by serving less video segments relative to other nodes in order to balance the load that the additional messages are responsible for. This will be considered by our future work.

D. Multipath Data Delivery

In order to improve QUVoD's data delivery performance, we designed a multipath data transfer mechanism complying

Algorithm 2 Video segment seek and download algorithm for vehicle V_i

```

1: /*  $V_i$  requests segment  $s_g$  */
2: /*  $|GMT|$  is the number of items in  $GMT$  */
3: for ( $forth = 0$ ;  $forth < M$ ;  $forth++$ )
4:    $fore\_limit = L_{group} \times (g - forth - 1)$ ;
5:   find  $fore\_limit$ 's Successor  $V_j$  by Chord algorithm;
6:   send a Query message to  $V_j$ ;
7:    $min = \infty$ ;  $vid = 0$ ;
8:   for ( $i = 1$ ;  $i < |GMT|$  of  $V_j$ ;  $i++$ )
9:     if ( $s_g$  is local stored &&  $V_j.GMT[i].Load < min$ )
10:       $min = V_j.GMT[i].Load$ ;
11:       $vid = V_j.GMT[i].NID$ ;
12:     end if;
13:   end for ;
14:   if ( $vid > 0$ ) break; end if
15: end for;
16: if ( $vid = 0$ )
17:    $V_i$  downloads  $s_g$  from media server employing multipath delivery
     mechanism; exit;
18: end if
19:  $V_{vid}$  sends a Query Reply message to  $V_i$ ;
20:  $V_i$  downloads  $s_g, s_{g+1}, \dots, s_{g+M_i-1}$  from  $V_{vid}$  employing the
     multipath delivery mechanism;
21:  $V_j$  updates  $V_j.GMT[i].Load$ ;
22: if (downloading  $s_t (t \in [g, g + M_i - 1])$  fails)
23:    $V_j$  selects another  $V_l$  with low stress in  $V_j$ 's  $GMT$  as  $V_i$ 's supplier;
24:    $V_i$  downloads segments between  $[s_t, s_{t+M_k-1}]$  from  $V_l$  employing
     multipath delivery mechanism;
25: end if

```

with QUVoD multi-homed structure. We implement the novel Next Generation Network-Oriented Mobile Equipment (NGN-Oriented ME) architecture [33] into our QUVoD system which can enable one of the available interfaces according to the specified Quality of Service (QoS) of data.

The NGN-Oriented ME architecture can provide a dual-mode such as WiMAX/WLAN interfaces within the Physical layer and an adaptive QoS module dubbed QoS-Cross-IP module (QXIP) within the IP layer. With QXIP, NGN-Oriented ME distributes the packet over either WLAN or WiMAX depending on the packet type of service. This paper enhances the Dual-mode ME architecture [33] with an Intelligent Data Distributor (IDD) integrated into QXIP. Fig. 6 shows the IDD-enhanced Dual-mode ME architecture.

Supposing a vehicle V_i wants to download video segments from a vehicle V_j and there are two available paths through V-Path and G-Path, the proposed IDD aims to schedule data distribution over V-Path and G-Path according to their current estimated Available Bandwidth (*AB*). V_i launches IDD to estimate *AB* of both V-Path and G-Path, following a well-known *AB* model for video transport [34] which is shown in equation (6).

$$AB = \frac{const}{RTT \times \sqrt{PLR}} \quad (6)$$

In equation (6) *const* is a constant with value of 1.22 or 1.33 and *PLR* can be evaluated by the two-state discrete Markov Chain known as Gibert's Model which is detailed in [34]. *RTT* can be estimated by equation (7).

$$RTT = \alpha \times \overline{RTT} + (1 - \alpha) \times (t - T_{send} - \Delta T) \quad (7)$$

In equation (7) \overline{RTT} denotes the current round trip time of path, t is the timestamp at the time at which the packet ACK

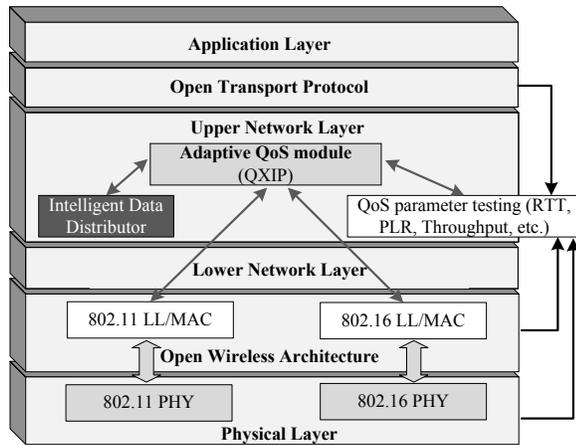


Fig. 6. The enhanced Dual-mode ME architecture with Intelligent Data Distributor.

is received at the sender, α is a weighting parameter with a common value of 0.875, T_{send} is the timestamp at the packet sending time, and ΔT is the time interval of a packet handling time at the receiver (it can be omitted in general).

V_i probes for V-Path's AB (AB_v) and G-Path's AB (AB_g) periodically (e.g. per RTT). The V-Path is preferred to distribute packets in VANET. However, if AB_v is less than a threshold AB_{thres} , the gap $AB_{gap} = AB_v - AB_{thres}$ is greater than a specified value AB_{value} , and in that case, AB_g is higher than AB_{thres} , V_i will distribute packets over G-Path. Otherwise, V_i will enable V-Path to send packets to V_j .

E. Speculation-based Pre-fetching

For a vehicle, in order to smoothen passengers' video playback, the video segments need to be pre-fetched and downloaded in advance into the vehicle's buffer. Pre-fetching has widely been used in current P2P VoD systems [5], [6], [9], [11] and [12]. However most of them only pre-fetch the subsequent video segment for sequence playback. When a user employs a jump operation, the algorithm should locate the video segment corresponding to the new position, fetch it and then play it back. The smooth playing will stop if long latency is encountered. So in interactivity cases, fast resource relocation and transmission is important, but not sufficient.

Unlike most of the nodes in other wireless networks, in VANETs vehicles can be equipped easily with **large-capacity-storage** and **powerful-computing devices**. For example, during the duration of a trip by taxi, passengers make up a captive audience and large quantities of data can be stored in viewing logs. Each vehicle in QUVoD periodically performs statistical analysis of the characteristics of user viewing behavior. Based on this analysis, when a user is playing the current video segment, QUVoD proactively pre-fetches the segment that is likely to be requested by either sequential playback or VCR-like operations.

A viewing log is actually a user's viewing trace in terms of these video segments and is recorded by the vehicle. Table I shows k user viewing logs, for the same video, identified by a unique Resource ID. The segments in each log are

TABLE I
EXAMPLE OF VEHICLE VIEWING LOG FOR THE VIDEO IDENTIFIED BY RESOURCE ID

Log ID	Playback Records
1	1, 2, 3, 7, 8, 9, 11, 12, 13, 16, 17, 20
2	1, 4, 5, 6, 7, 10, 11, 12, 13, 16, 17, 18, 20
3	1, 2, 3, 4, 5, 9, 10, 12, 13, 16, 17, 6, 7
4	1, 2, 3, 8, 9, 12, 13, 14, 16, 17, 18, 9, 10
...	...
k	1, 2, 3, 4, 5, 9, 10, 11, 12, 13, 16, 17, 19, 20

sequence-sensitive. Existing studies employ statistical analysis of viewer playback records to calculate segment popularity in order to support interactivity-related business [9], [11] and [12]. However, most of the studies neglect the association between segments. In this paper, we will consider segment-based and substring-based association to predict and pre-fetch the video segment that is most likely expected by the users. This approach is meant to significantly smoothen user video playing experience in VANETs. Furthermore, we also employ a feedback-based self-regulation mechanism to dynamically regulate the association probability.

Segment-based Association:

In terms of the sequence of segments, the likely playback association between segments can be obtained for example as follows: $1 \rightarrow 2$, $1 \rightarrow 4$ and $2 \rightarrow 3$. This is as user interactivity often breaks the sequential order of segments, namely each segment s_i can be associated with $n-1$ segments (if the jump target is within the current video segment, the jump behavior is called "inner-segment jump" and does not require special treatment). Our solution counts the frequency of the association between each segment s_i and the other $n-1$ video segments. For instance, in table I, the frequency of the associations $1 \rightarrow 2$ and $1 \rightarrow 4$ is 3 and 1 in the first four logs, respectively. A matrix containing the frequencies of the associations for all segments in the k entry log can be defined as:

$$\begin{pmatrix} f_{11} & f_{12} & \cdots & f_{1n} \\ f_{21} & f_{22} & \cdots & f_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ f_{n1} & f_{n2} & \cdots & f_{nn} \end{pmatrix}$$

In the matrix, the value of the main diagonal should be set 0 due to the fact that inner-segment jumps is in fact local seeking so it is not considered as an association. The total frequency of the associations between s_i and the other $n-1$ segments is in fact the sum of the frequency entries in the row corresponding to s_i and is expressed in equation (8).

$$SUM(s_i) = \sum_{c=1}^n f_{ic}, \quad SUM(s_i) > 0 \quad (8)$$

If the users have accessed any segment s_i , the sum of the i^{th} row in the matrix should be greater than 0. Considering the frequency of the association between segment s_i and any segment s_j , denoted f_{ij} and that $SUM(s_i) > 0$, the jump probability from s_i to segment s_j is computed according to equation (9).

$$P_{ij}\{s_j|s_i\} = \frac{f_{ij}}{SUM(s_i)} \quad (9)$$

Moreover, the frequency of s_i to s_j jumps in the k -entry log can be considered as a weight factor. The higher the frequency of accessing s_j from s_i is, the more accurate the probability of the $s_i \rightarrow s_j$ association is. The weight value of s_i can be obtained according to equation (10).

$$w_{s_i} = \frac{r_{s_i}}{\sum_{c=1}^n r_{s_c}} \quad (10)$$

where r_{s_i} is the overall frequency of s_i in k logs. $\sum_{c=1}^n r_{s_c}$ indicates the overall frequency of all segments in the k -entry log. Next we can obtain the weighted probability of the $s_i \rightarrow s_j$ association according to equation (11).

$$WP_{ij}\{s_j|s_i\} = w_{s_i} \times w_{s_j} \times P_{ij} \quad (11)$$

The weighted probability for the associations between each segment to other segments can be therefore obtained. Based on the matrix of association frequencies we can build the matrix of weighted probability associations as follows.

$$\begin{pmatrix} WP_{11} & WP_{12} & \cdots & WP_{1n} \\ WP_{21} & WP_{22} & \cdots & WP_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ WP_{n1} & WP_{n2} & \cdots & WP_{nn} \end{pmatrix}$$

In the matrix, WP_{ij} represents the association probability between two segments i and j only, and cannot reflect the behavior of the continuous movement of the playback point, which most of the time includes a large series of segments. Therefore we also use a substring-based weight factor to further influence the segment-based association probability.

Substring-based Association:

The continuous segments form a substring in their playback order. For example there are five substrings: $\{1, 2, 3\}$, $\{7, 8, 9\}$, $\{11, 12, 13\}$, $\{16, 17\}$, $\{20\}$ in the first entry in the k -entry log $[1, 2, 3, 7, 8, 9, 11, 12, 13, 16, 17, 20]$ in table I. Even if a single segment playback occurs in a log, it is also considered as a substring such as the case for 20. Next we obtain a substring set by extracting the substring sample from the k -entry log, namely $strSet \Leftrightarrow (str_1, str_2, \dots, str_m)$. Similar to the segment-based matrix of weighted association probability, we can obtain the substring-based matrix of weighted probability associations, as follow.

$$\begin{pmatrix} WP_{11}^{(s)} & WP_{12}^{(s)} & \cdots & WP_{1m}^{(s)} \\ WP_{21}^{(s)} & WP_{22}^{(s)} & \cdots & WP_{2m}^{(s)} \\ \vdots & \vdots & \ddots & \vdots \\ WP_{m1}^{(s)} & WP_{m2}^{(s)} & \cdots & WP_{mm}^{(s)} \end{pmatrix}$$

Moreover, m , the length of $strSet$ ($m \leq \sum_{1 \leq t \leq n} \left[\frac{t^n}{t!} \times \sum_{0 \leq e \leq n-t} \frac{(-1)^e}{e!} \right]$) is variable as new substrings may be generated at any time.

Algorithm 3 Speculation-based pre-fetching strategy

```

1: //Let  $s_i$  be current playback segment;
2: //Let  $s_r$  and  $s_p$  be real and predicted target segment;
3: get probability set  $(WP_{i1}, WP_{i2}, \dots, WP_{in})$  by equation (11);
4: get weighted probability set  $CPS_i$  by equation (12);
5: select  $s_p$  with  $MAX(CPS_i)$  as the predicted segment;
6: if ( $s_p == s_{i+1} || s_p == s_{i-1}$ )
7:   locate  $s_p$  by using NSL or PLS;
8:   pre-fetch  $s_p$  by employing multipath delivery mechanism;
9: else
10:  pre-fetch  $s_p$  by algorithm 2;
11: end if
12: if ( $s_r == s_p$ ) regulate  $\alpha_{ij}$  with  $I(+)$ ; end if //achieve smooth playback;
13: else if ( $s_r == s_{i+1} || s_r == s_{i-1}$ )
14:  seek  $s_p$  by NSL or PSL;
15:  else seek and download  $s_p$  by algorithm 2;
16:  end if
17:  regulate  $\alpha_{ij}$  with  $I(-)$ ; //prediction fails;
18: end if

```

Assuming that the playback point has gone through h sequential segments before accessing s_i and is located into s_i , namely there is a substring $str_{t(i)} = (s_{i-h}, s_{i-h+1}, \dots, s_i)$, we use the accumulated value of substring-based probability from substring $str_{t(i)}$ to all substrings whose first segment is s_j to indicate the substring-based weight factor. The substring-based weighted combinatorial probability can be then calculated according to equation (12).

$$D_{s_i \rightarrow s_j} = (WP_{ij})^{\alpha_{ij}} \times \sum WP_{t(i)c(j)}^{(s)} \quad (12)$$

where α_{ij} ($0 < \alpha_{ij} \leq 1$) is the impact factor of WP_{ij} , used to regulate WP_{ij} 's contribution according to the prediction results for pre-fetching. WP_{ij} 's value can be adjusted in real time based on the user playback interest variation, instead of being updated after collecting large amounts of log data and recalculation, as in the classic approach. α_{ij} will be further discussed next. $WP_{t(i)c(j)}^{(s)}$ is the association probability between $str_{t(i)}$ and $str_{c(j)}$ whose first segment is s_j . $\sum WP_{t(i)c(j)}^{(s)}$ is the substring-based weight factor. Next we obtain the combinatorial probability set from s_i to the other $n-1$ segments, namely $CPS_i = (D_{s_i \rightarrow s_1}, D_{s_i \rightarrow s_2}, \dots, D_{s_i \rightarrow s_n})$. Therefore, the target segment in terms of what user wants to view next is s_j with probability $D_{s_i \rightarrow s_j} = MAX(CPS_i)$.

The prediction of playback behavior aims to ensure users' quality of viewing experience.

(1) As often the target segment predicted according to the above-indicated algorithm is the next or previous segment of the current segment, similar to VMesh, a peer in QUVoD keeps two lists of peer pointers, namely next-segment-list (NSL) and previous-segment-list (PSL). By making use of NSL and PSL, the user can connect with the suppliers in overlay 1-hop and pre-fetch the video segment immediately.

(2) If the target segment predicted is not the next or previous segment of the current segment, the user seeks the predicted segment and fetches it in advance. If the current available bandwidth permits, the next segment is also pre-fetched as well. Based on this, smooth playback supported whether sequential playout or jumps occur in practice.

With the increase in the viewing log entries, the statistics of segments sequence and substrings continually updates so that the corresponding probability also continually changes, improving its accuracy. However these frequencies are updated regularly every period of time UT set in such a manner not for the real-time updates to determine high computation costs. This efficiency design decision does not affect the computation accuracy of the probabilities too much, as few new viewing log entries do not influence too much the overall values of these probabilities. After every period UT , each node combines the incremental and history statistics to obtain the new probabilities. For instance, let INC_{ij} be the frequency increment for the association from s_i to s_j . We can obtain the new probability by modifying equation (9) to equation (13). By this way, the computational complexity significantly reduces.

$$P'_{ij}\{s_j|s_i\} = \frac{f_{ij} + INC_{ij}}{SUM(s_i) + INC_{ij}} \quad (13)$$

The statistics-based prediction scheme depends on the confidence level of the available heuristic information (viewing log). If the predicted s_j is the real segment user is requesting after playing s_i , it means the prediction is accurate and, the weighted probability of the $s_i \rightarrow s_j$ association can be increased, otherwise it should be reduced. We use the feedback information on the prediction accuracy to adjust the impact factors α_{ij} in equation (12). The new α'_{ij} can be updated according to the equation (14):

$$\alpha'_{ij} = \begin{cases} \alpha_{ij}^{(1+\rho)}, & I(+), \\ \alpha_{ij}^{(1-\rho)}, & I(-) \end{cases}, 0 < \rho < 1 \quad (14)$$

where ρ is global variable and considered as feedback factor. The $I(+)$ and $I(-)$ represent positive and negative feedback respectively, namely the success or failure of prediction. The above Reinforce Learning model of α_{ij} is to improve the accuracy of prediction by exploiting the user's feedback information. Algorithm 3 describes the speculation-based pre-fetching strategy.

IV. PERFORMANCE EVALUATION

A. Simulation Methodology

QUVoD's performance is assessed in comparison with that of VMesh, in VANETs by making use of the Network Simulator (NS2). The application-layer Chord network of both QUVoD and VMesh are implemented based on the MChord [19]-[20], extended according to their different individual algorithms. For QUVoD, we consider two wireless communication interfaces for each of the vehicle nodes: the IEEE 802.11p WAVE network interface and 4G WiMAX interface. The experiments are implemented using the Multi-interface Cross Layer extension for NS2 (NS-MIRACLE) [35] as it enables the co-existence of multiple modules within each layer of the protocol stack, supporting the integration of IEEE 802.11p and WiMAX network interfaces. The IEEE 802.11p and WiMAX libraries of the NS-MIRACLE are available at [36]. For IEEE 802.11p, two WAVE channels are used at PHY/MAC layer. One is a control channel (CCH) which is

TABLE II
SIMULATION PARAMETERS SETTING FOR VANET AND 4G

	Parameters	Values
VANET	Area	2000 × 2000(m ²)
	Channel	Channel/WirelessChannel
	Network Interface	Phy/WirelessPhyExt
	MAC Interface	Mac/802_11Ext
	Peak Data Rate	11 Mbps
	Frequency	5.9 GHz
	Multiple Access	OFDM
	Channels Sharing	CSMA/CA
	Transmission Power	33 dBm
	Wireless Transmission Range	250m
	Interface Queue Type	Queue/DropTail/PriQueue
	Interface Queue Length	50 packets
	Antenna Type	Antenna/OmniAntenna
	Routing Protocol	AODV
	Peak Mobility Speed	30 m/s
	Mobility Model	Manhattan Mobility Model
Transport-Layer Protocol	TCP	
4G WiMAX	Operating Frequency	2.5 GHz
	Channel Bandwidth	10 MHz
	Channel Capacity	40 Mbps
	Radio Propagation Model	MPropagation/MFreeSpace
	Frame Duration	5ms
	Multiple Access	OFDM
	Duplexing Mode	TDD
	Interface Queue Length	50 packets
	Cell Layout (BS Number)	7 hexagonal cells
	Cell Radius (Transmission Range)	578m
Transport-Layer Protocol	TCP	

employed for delivering control information. The second one is a service channel (SCH) which is used for delivering data.

To reflect the realistic mobility characteristics of VANETs, we use SUMO [37] to generate street scenarios. A street area of 2000 × 2000(m²) which consists of five horizontal streets and five vertical streets is used in the experiment. As Fig. 7 (a) shows, every street has two lanes in each direction and the Manhattan Mobility model [38] is employed for vehicular movement. The vehicular nodes are uniformly distributed over the streets and move along the directions shown by the arrows. Whenever a vehicle reaches an intersection, it is determined with some fixed probability whether it would keep moving in the same direction with 50% probability or turn left or right with 25 % probability in each case. The 4G WiMAX layout (Fig.7 (b)) is composed of 7 hexagonal cells with 578m for the cell radius, namely 7 numbers of base stations(BSs) with 578m transmission range. We consider a scenario in which maximum 1000 vehicles request videos. Around 143 nodes connect and share the same downlink/uplink bandwidth available for each 4G base station. BSs are connected through a 128 Mbps wired link with a 2ms delay. The horizontal handover of vehicle nodes between BSs adopts the Hard Handover (HHO) mechanism in IEEE 802.16e [39]. This mechanism has been implemented in NIST module [40] as extension of NS2 and also has been integrated in NS-MIRACLE framework by University of Karlstad [41]. Tables II lists some important NS2 simulation parameters of the VANET and 4G WiMAX networks, respectively. Considering multipath data delivery, we set the value of AB_{thres} to 128 kbps which equals the streaming rate, and the AB_{value} is 8 kbps in our experiments. Namely, if the value of V-Path's AB (in equ. 6) is less than 120 kbps and G-Path's AB is greater than 128 kbps, the packets will transfer over the G-Path. Otherwise, the V-Path is used to deliver the video segments.

As VMesh is a single-tier solution, we consider two P2P

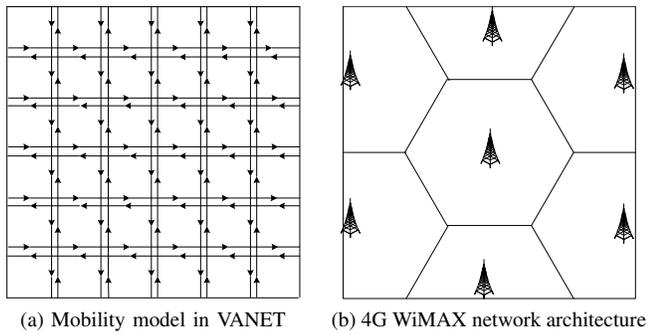


Fig. 7. VANET and 4G simulation scenarios.

overlay construction ways for VMesh. One is VMesh(ah-hoc), in which vehicles form the Chord on top of VANET. The other case is VMesh(4G), in which the Chord is established over the 4G infrastructure. In VMesh(ad-hoc), packets transfer through V-Path via the WAVE interface. VMesh(4G) uses the G-Path via the WiMAX interface for data transfer, respectively. As in VANET mobile nodes' position changes continually, VMesh's locality-aware segment management is unusable. Instead, the simulations use the more common case of VMesh which applies random selection for segment storage.

Simulations consider a 5400 seconds long video which is divided into 180 segments. The streaming rate is 128 kbps. Each segment is 0.5 minutes long and about 0.47 MB in size. In QUVoD, the maximum storage capacity parameter M is set to 5. As discussed in Section III.E, by analyzing user viewing logs data, QUVoD applies the speculation-based strategy to adapt to the users' interactive behavior. First we created 5000 synthetic user viewing log entries based on the interactive actions according to the measurements and statistics from [42]. The video was one of the most popular matches in terms of the number of viewers. The impact factor α_{ij} in equation (12) was set 1 to make the segment-based association and substring-based association have equal impact on the prediction. By using the same method, we then generate 1000 sole user viewing logs and assign each of them to each of the 1000 nodes to create the multiple requests scenario. The playback trace of a node can be determined by its assigned viewing record. For instance, a vehicle with playback record (1, 2, 3, 4, 7, 8, 9, 10, 14, 15, 16, 17, 18, 40, 28, 29, 30, 31, 44, 45, 46, 47, 52, 53, 60, 61) has a lifetime of 13 minutes and takes seven VCR operations in total.

We compare QUVoD with VMesh(ad-hoc) and VMesh(4G) in terms of the user QoE-related indicators including average lookup success rate, average segment seeking latency and hit ratio in supporting interactivity, and overhead-related metrics including vehicles stress distribution, media server stress and system control overhead.

B. Simulation Results

1) Average lookup success rate (ALSR): The video segment ALSR describes system serving capability. ALSR is defined as the ratio between the number of times lookup success was obtained and the total number of lookup tries. Both VMesh and QUVoD are P2P-based VoD solutions and

in order to fairly compare the efficiency of their P2P-related algorithms including video segments storage and seek strategy *etc.*, we neglect the requests from the server for computing ALSR. High ALSR indicates it is easier for a vehicle node to obtain video resources from other suppliers instead of from the media server. This improves system's resources sharing capability and indirectly reduces the media server stress. The experiments run involve variation in the mobility speed of VANET vehicles and different vehicular node numbers.

Fig. 8 shows the ALSR variation with the increase in the number of vehicles from 200 to 1000, where the mobility speed varies in the $[0, 30]$ m/s range. As the figure shows, with the system scale increase, ALSRs of QUVoD and VMesh all increase. This is as increasing number of vehicles determines higher segment sharing chances and any vehicle can find easier a resource supplier. ALSRs of QUVoD and VMesh increase quickly when the vehicles number increases from 200 to around 500. Above 600 nodes, ALSRs of QUVoD and VMesh maintain a relatively stable value close to 1. However, when system scale is low (there are less than 700 nodes), ALSR of QUVoD is higher than that of VMesh(4G). This is because QUVoD's grouping-based segments storage enhances ALSR compared with the random segments storage used in VMesh. ALSR of VMesh(ad-hoc) increases slowly and only arrives at about 0.645 when the number of vehicles increases to 1000. QUVoD and VMesh(4G) locate the video segments over Chord by using the underlying 4G network which is stable and reliable. The vehicles' movement in a vehicular scenario does not influence the Chord circle. As long as there are enough vehicles, a supplier can always be found via 4G by DHT. However as VMesh(ad-hoc) is formed on top of VANET, ALSR is easily affected by vehicle density, especially in low-density situations. As the figure shows, when there are 1000 vehicles, ALSRs of QUVoD and VMesh(4G) are about 35% higher than ALSR of VMesh(ad-hoc). However, the difference between ALSRs can be higher than 70% when the number of nodes is 600.

Fig. 9 illustrates ALSR comparison results with the variation in the mobility speed of VANET vehicles. In order to minimize the impact caused by the number of nodes, the number of vehicular nodes is set to 1000. We can see in the figure that the ALSRs of QUVoD and VMesh(4G) always maintain high levels. ALSR of VMesh(ad-hoc) decreases with the increase of mobility speed variance. This is as in VMesh(ad-hoc), the increase in vehicle mobility, determines the P2P overlay over the mobile ad-hoc network to unstable, which results in some lookups not to reach the vehicles with requested segments. Unlike VMesh(ad-hoc), message communications in QUVoD and VMesh(4G) are through the more stable 4G network and the increase in the movement of vehicles can hardly influence data communication.

2) Average segment seeking latency (ASSL): ASSL is the average time from a vehicle requesting a video segment to having received the segment into its playback buffer. ASSL mainly includes the average segment locating latency and average segment downloading latency. When a vehicle joins the system, ASSL determines how long a user needs to wait for the download of initial video segments to begin playing, so

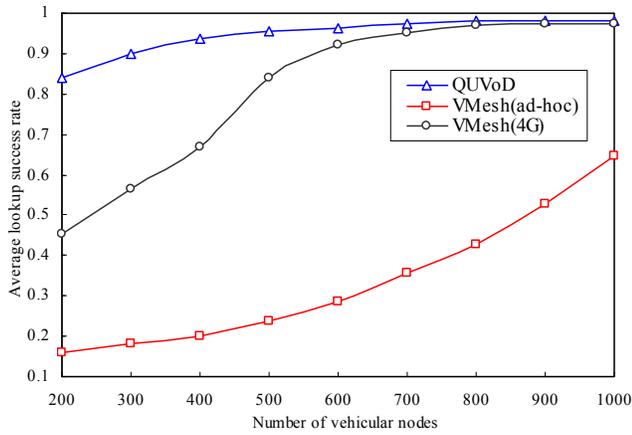


Fig. 8. Average lookup success rate against number of vehicular nodes.

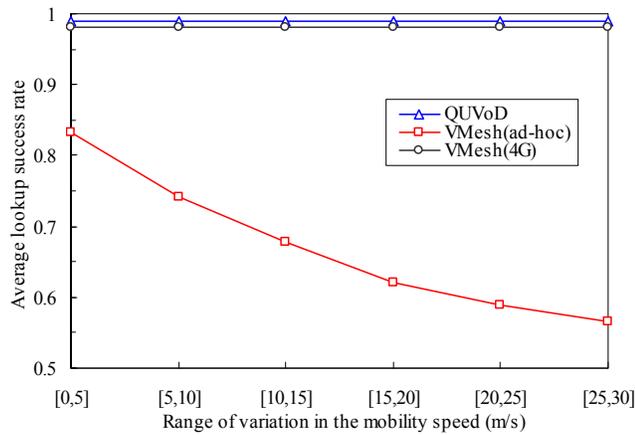


Fig. 9. Average lookup success rate against different vehicle mobility speed variation.

ASSL has a significant impact on joining and startup latency. In addition, during video playing process, long ASSL can severely affect user viewing QoE levels as late arriving data may cause buffer underflow. Consequently there is a desire to have short ASSL. In simulations, we first compute the *average segment locating latency* in the Chord network, and then compute the *average end-to-end transfer delay* for a segment between two vehicle nodes. The sum of these two values determine ASSL. As vehicles may request segments many times, only the success in the seeking process will be considered in the calculation of the average ASSL.

Fig. 10 plots the comparison results when QUVoD and VMesh are used in turn for increasing number of vehicular nodes, where the mobility speed varies in the $[0, 30] m/s$ range. As the figure shows, ASSL of VMesh(ad-hoc) is higher than that of QUVoD and VMesh(4G), especially when the system scale is low (vehicles number is less than 400). This is because low density of nodes in VANET reduces connectivity between nodes. QUVoD performs better than other two methods. For example, in the case of 200 vehicles, QUVoD's ASSL is about 0.573 seconds, 7% lower than that of VMesh(4G) (0.613 seconds) and 36.8% lower than that of VMesh(ad-hoc) (around 0.784 seconds). Fig. 11 illustrates ASSL comparison results for different ranges of mobility speeds where the number of

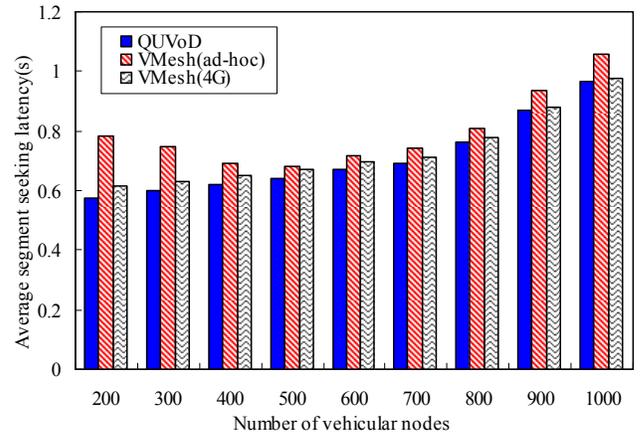


Fig. 10. Average segment seeking latency against number of vehicular nodes.

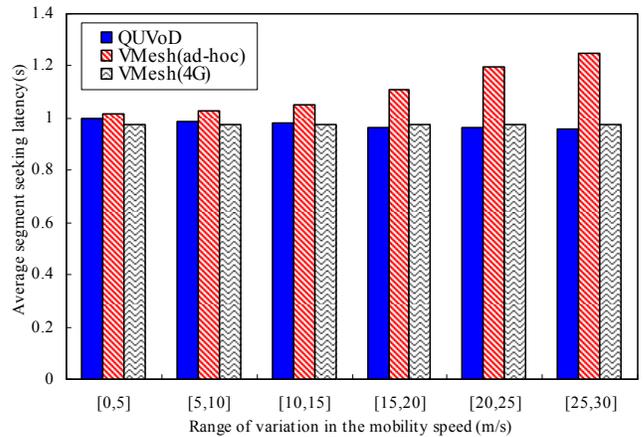


Fig. 11. Average seeking latency against mobility speed of vehicles.

vehicular nodes is set to 1000. As the figure shows, when nodes have low mobility speeds (less than $10m/s$), VMesh(ad-hoc) performs well. However with increasing mobility speeds, VMesh(ad-hoc) performs the worst among the three methods compared. QUVoD's ASSL is slightly higher than that of VMesh(4G) in the low mobility speeds case and the reason is that QUVoD mainly uses V-Path for data delivery which can be negatively impacted by V2V communications. However the difference is very small and in all other cases ASSL of QUVoD is maintained at a good level. This is as QUVoD stores several sequential segments in one node, the segments excluding the first segment do not need to be located again, which significantly reduces *average segment locating latency* and mitigates the negative impact brought by the partial use of V-Path.

In VMesh(ad-hoc), both the search message and data are transmitted through the VANET network which is unreliable, so ASSL and ALSR of VMesh(ad-hoc) is worse than those of QUVoD and VMesh(4G) in low density and high mobility speed situations. VMesh(4G) shows a relatively good performance for both ALSR and ASSL. However, the network usage cost of VMesh(4G) is the highest as all the search messages and data transfers are performed over the 4G network. For QUVoD, the search messages transfer in the 4G network and

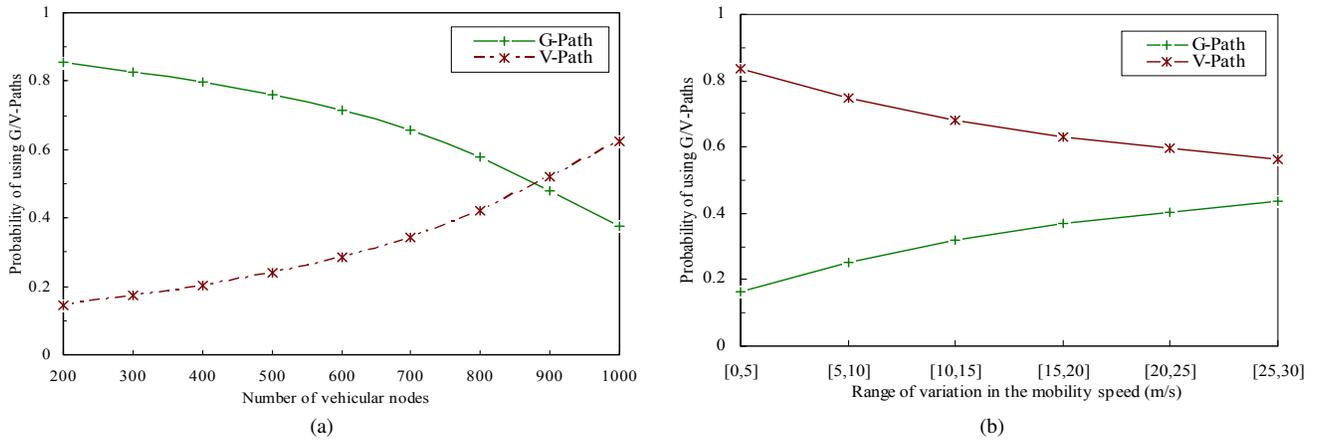


Fig. 12. Evaluation on probability of using G/V-Paths for downloading segments.

the video data is based on the multipath transfer in multi-homed 4G/VANET network, so QUVoD has both high seeking success rate and relatively low segment seeking delay.

3) G/V-Paths evaluation: Fig. 12 illustrates the investigation on the average probability of using G-Path and V-Path to download video segments by applying multipath data delivery mechanism in QUVoD. Fig. 12 (a) investigates the probability against the increase in the number of vehicular nodes, where the mobility speed varies in the $[0, 30]$ m/s range. As it is shown, when there are few vehicles in the system, for instance 200 vehicles, the probability of employing G-Path is around 85.6% for segment download and 14.4% for using V-Path. However, as the system scale increases, the probability of using G-Path significantly decreases. For example, when system scale reaches 1000 nodes, the probability of using G-Path is only 37.5% and V-Path's usage probability reaches 62.5%. Fig. 12 (b) evaluates the probability of using G/V-Paths in terms of range of variation in the mobility speed. It illustrates how for slow speeds, the probability of using G-Path is also small (e.g. when the speed is in the $[0, 5]$ m/s range, the probability of using G-Path is 16.5% and that of employing V-Path is 83.5%). However, with speed increases to belong to the $[25, 30]$ m/s range, the probability of using G-Path increases to 43.8% and that of V-Path decreases to 56.2%. Fig. 12 fully illustrates that QUVoD can combine the merits of VANET and 4G network, and achieve a desirable tradeoff between quality and network usage cost.

4) Average hit ratio (AHR): We use the AHR metric to evaluate the performance in terms of supporting interactivity. After finished playing a segment, a vehicle then continues seeking a next segment for play-out. It will check first its playback buffer. If the buffer contains the required segment, it is considered as a *hit* event and the vehicle can continue video playback smoothly without any suspension. Otherwise, it is seen as a *miss* event and the vehicle needs to relocate and fetch the required segment, which involves long latency and seriously deteriorate user QoE level. The average hit ratio is the ratio between the total number of hit events and the total number of seeking requests. In this experiment, we investigate the average hit ratio when the number of vehicles increases. The mobility speed is in the $[0, 30]$ m/s range.

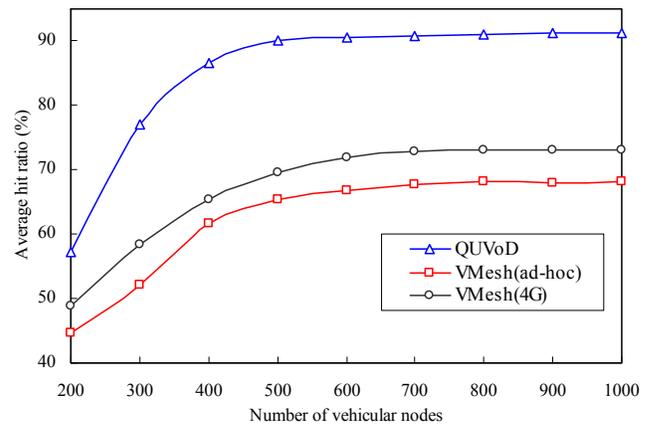


Fig. 13. Average hit ratio against number of vehicular nodes.

As Fig. 13 shows, if the system has small-scale, increasing number of vehicles can improve AHR. However, increasing the vehicle number beyond 600, sees almost no effect on the AHR values of QUVoD and VMesh. This is because the P2P sharing capability is no longer the influencing factor for AHR; the segment pre-fetching scheme has become the dominating factor for AHR. We can see that AHR values of QUVoD are about 20% and 25% higher than those of VMesh(4G) and VMesh(ad-hoc), respectively. In VMesh, only sequential playback behavior determines a hit and increases the AHR; any interactivity determines pre-fetching to fail which decreases the AHR. However in QUVoD, by employing speculation-based pre-fetching strategy, the segments which are not only for sequence playback but also for jump can be pre-fetched with high possibility, which make QUVoD achieve higher AHRs and better QoE than VMesh. Fig. 13 also shows how around 10% of pre-fetching fails in QUVoD, which means some pre-fetched segments cannot be used for playback. Although this wastes some additional bandwidth and storage for those failed pre-fetching, considering the increasing bandwidth on VANETs and storage capability on local vehicles, it actually offers a very desirable tradeoff between quality and cost.

The Fig. 14 (a), (b) and (c) show the average AHR distribution of QUVoD and VMesh related to the seeking event between segments, respectively. The average AHR of each

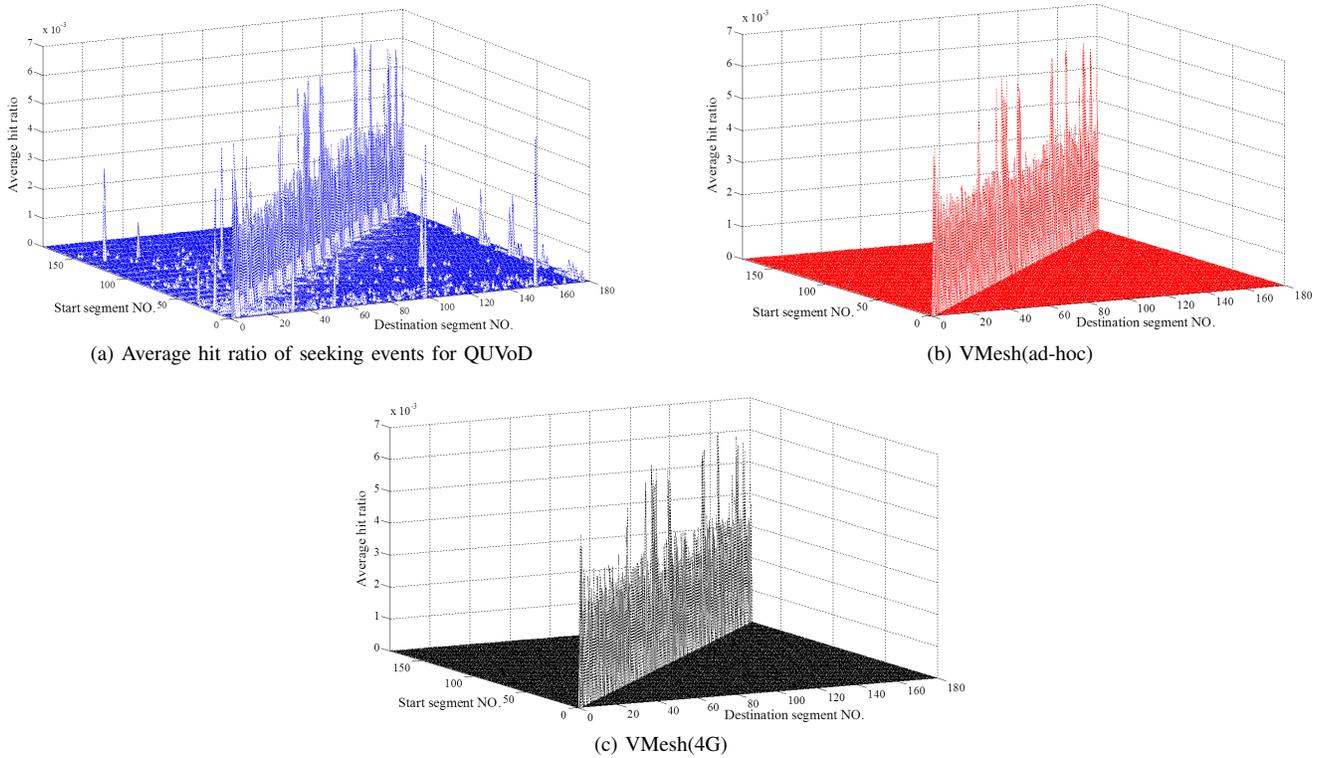


Fig. 14. Average hit ratio against seeking events.

seeking event $S_{s_i \rightarrow s_j}$ is defined as:

$$HR_{ij} = \frac{f_{ij}^H}{\sum_{b=1}^n \sum_{c=1}^n f_{bc}}, \quad f_{ij}^H \leq f_{ij}, \quad HR_{ij} \in [0, 1] \quad (15)$$

where f_{ij} and f_{ij}^H are the occurrence frequency of $S_{s_i \rightarrow s_j}$ and its *hit* event, respectively. $\sum_{b=1}^n \sum_{c=1}^n f_{bc}$ is the occurrence frequency of all seeking events. As Fig. 14 shows, AHR values of VMesh are all distributed in a plane which is composed of coordinate axis Z and a straight line closed to leading diagonal in the plane XOY. This is because the users in VMesh pre-fetch sequentially the next video segment only. The AHR values of VMesh(4G) are greater than those of VMesh(ad-hoc) due to the stability of the 4G network. In general, the AHR values of QUVoD are far greater than those obtained by VMesh. QUVoD can provide accurate prediction results for all seeking events to ensure the high playback continuity with the help of the proposed speculation-based pre-fetching strategy.

5) Vehicles stress: In QUVoD and VMesh, the vehicular nodes not only need to seek and download new video content from other vehicular nodes after consuming the current video segment, but also respond to seek requests from other nodes and deliver the streaming data requested. We investigate the distribution of the number of vehicular nodes relative to the number of seeking events and corresponding Cumulative Distribution Function (CDF) to support the analysis of vehicle stress, respectively. In Fig. 15 (a), the curve corresponding to the distribution experiences first a rise and then a fall, with the peak value in the range [120, 140]. The number of seeking

events is in the range [56, 189] and the average number of seeking events for each vehicular node is 128.97. Fig. 15 (b) shows the CDF corresponding to the distribution in Fig. 15 (a). As we know that the number of seeking events is greater than the total number of video segments (180), there is a possibility that some users watched the entire video. In Fig. 15 (b), the curve in the range [180, 200] tends to a smooth horizontal line (CDF is close to 1), indicating that a small minority of users could have possibly viewed the whole video. The curve fast rises in the range [100, 160], which means that the number of seek events is in this range for most users.

The stress (load) distribution among the vehicular nodes in our simulation is a statistics about the number of vehicular nodes in different stress states. This is computed as the ratio between each node's stress value and the total stress of all vehicular nodes, where the number of served segments denotes the stress of each vehicular node. Fig. 16 illustrates the comparison statistics about the number of nodes in different ratio ranges between VMesh and QUVoD, where the total number of vehicular nodes is 1000. As the figure shows, more than 15% nodes in VMesh do not bear any load from start to end. They just download segments from others, but do not supply any resources. In QUVoD, each node bears a part of the system's load, and very few nodes are overloaded. It is important to note that the number of vehicles who bear load in the range (0, 2] exceeds 700 out of the total of 1000 nodes, which is a highly positive results for QUVoD.

In VMesh, the random storage scheme determines uneven video segment distribution in the Chord network, which leads to important load imbalance between nodes. However, in QUVoD, the grouping-based segment storage scheme enables

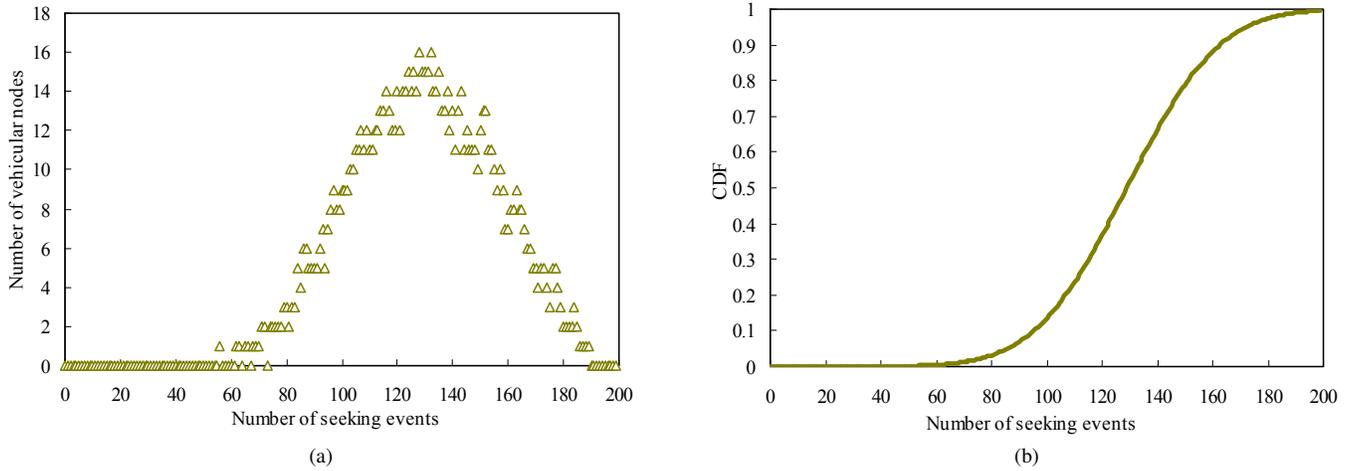


Fig. 15. Distribution and CDF of vehicular nodes against number of seeking events.

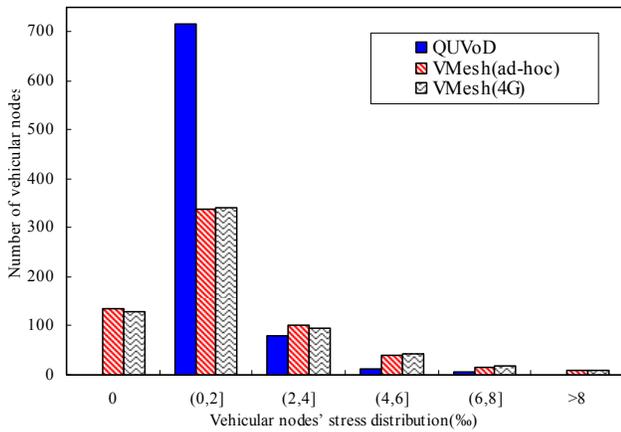


Fig. 16. Vehicular nodes' stress distribution.

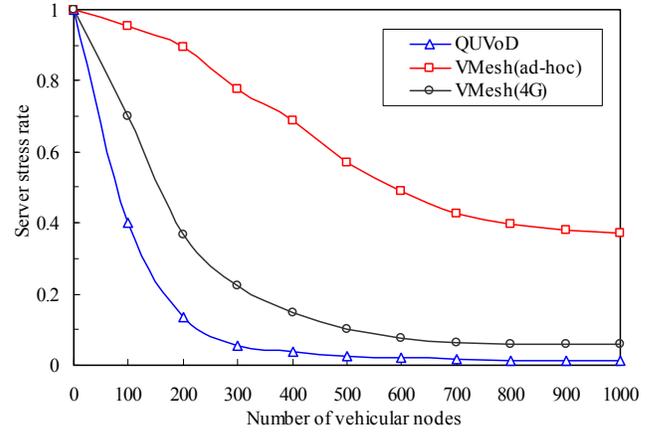


Fig. 17. Media server stress against the number of vehicular nodes.

uniform distribution of all the segments across the nodes. In general, S_{max} numbers of nodes can store all the video segments of the requested video. Furthermore, the nodes in the same group can serve the same video segment request and the nodes with low stress are preferred to provide service, which further helps achieve load balancing. In this way, each vehicle node in QUVoD shares a part of the load of the system.

6) Media server stress: Besides, in the simulation, we also investigate the stress on the media server. Media server is a backup resource provider; the vehicular nodes will download segments directly from this server when they cannot fetch the required resource from other nodes successfully. Obviously, the fewer streams the media server delivers, the lower the stress on the server is. Fig. 17 illustrates the server stress measured by the average ratio between the number of streams served from the server and the total number of streams in system during a period of simulation time. The vehicles number changes from 0 to 1000 with the mobility speed in variation range of [0, 30] *m/s*. We can see in the figure how the media server stress rate in VMesh and QUVoD decreases with the increase in vehicles number, as the sharing capability between vehicles increase with the system scale increase.

However, compared with VMesh, the server stress of QUVoD remains at low levels and becomes close to zero after about 600 vehicles. This lightweight load is due to the grouping-based distributed segments storage strategy and utilizing the multipath transfer mechanism. Due to the random storage strategy and also VANET's instable characteristics, the server stress of VMesh(ad-hoc) is in a high state. As Fig. 17 shows, it can reach at about 0.4 even when the vehicle number increases to 1000. The server stress of VMesh(4G) perform better than VMesh(ad-hoc) due to the stability merit of 4G network.

7) Control overhead: In QUVoD and VMesh, nodes joining, building and maintaining the Chord P2P network, and video segment seeking *etc.* require nodes to exchange control messages. We count all control messages as control traffic overhead and use the control messages' average occupied bandwidth per second as the control overhead. As shown in Fig. 18, the control overhead of QUVoD and VMesh grows with the increase in the number of vehicles. This is mainly as the increase in system scale will determine an increase in the number of exchanged control messages between nodes and will result in an increase in the Chord overlay maintenance cost. VMesh(ad-hoc) performs the worst as the Chord is

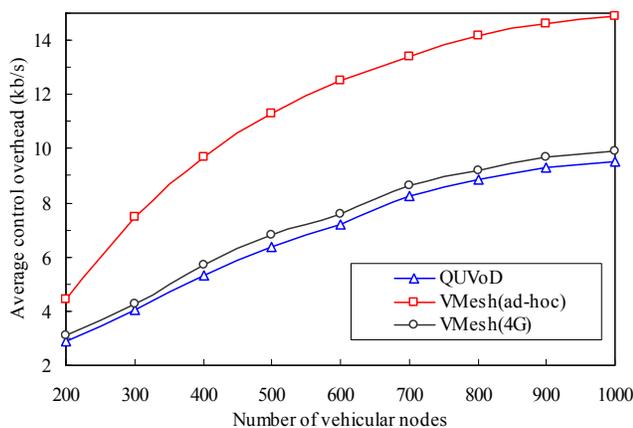


Fig. 18. Control traffic overhead.

formed directly on top of VANET, the overlay topology is easily affected by the underlying vehicles' mobility, and each vehicle needs frequent exchange of messages to maintain in real-time the connections for nodes frequently joining and leaving. However, as the Chord of QUVoD and VMesh(4G) are built over 4G network which is robust to variable network conditions, the topology maintenance messages can be significantly reduced. In QUVoD, because of the storage scheme by which we store several sequential segments in one node, more than one segment can be found at one time, which certainly greatly reduces the average seeking segments time; the control messages are also reduced correspondingly. Consequently, compared with VMesh(4G), QUVoD achieve lower control traffic overhead.

V. CONCLUSIONS AND FUTURE WORK

The very attractive user-centric mobile VoD service can support different applications in vehicular ad-hoc networks. It is a great challenge to support high quality of experience for interactive mobile VoD services, not only due to the high mobility of the devices, but also due to the users' interactive viewing behavior. In this paper, we proposed an efficient user-centric mobile VoD solution called QUVoD in urban vehicular network environment, which offers high QoE service level to vehicle passengers. Based on the proposed efficient designed multi-homed hierarchical P2P/VANET structure, four novel mechanisms are introduced: **distributed grouping-based video segments storage scheme**, **video segment seeking scheme**, **multipath data delivery mechanism** and **speculation-based pre-fetching strategy**. The storage scheme distributes the segments along the Chord network uniformly by groups allowing each video segment to be stored in multiple nodes, and each nodes to store multiple video segments. This not only balances nodes' load, but also reduces the resource seeking times. By employing the video segment seeking scheme, the video segments lookup messages transverse the Chord network by underlying 4G network. The stability of the 4G network ensures high lookup success rate. After lookup, the video segments are downloaded from the located node by the multipath data delivery mechanism over multi-homed 4G/VANET networks, achieving high data rate and reliability. Furthermore,

the presented pre-fetching strategy can sense user's viewing behavior and make intelligent decisions for fetching expected segments in advance, which greatly smoothen the playback experience for interactive users. The simulation-based results show how QUVoD is an efficient interactive mobile VoD solution in urban vehicular networks.

Extending the sophisticated multipath data delivery mechanism presented in this paper, future work will consider employing a Concurrent Multipath Transfer (CMT) strategy [30] and other CMT-related technologies in QUVoD, and distribute data across over G-Path and V-Path concurrently to achieve bandwidth aggregation. It can accelerate video segment fetching speed to further improve QUVoD's performance. In addition, we will also explore multiple video streams delivered simultaneously. Those results will be very useful and are expected to provide support for the very attractive high-definition (HD) VoD services in future urban multi-homed P2P-based vehicular networks.

REFERENCES

- [1] G. Piro, L. A. Grieco, G. Boggia, F. Capozzi and P. Camarda, "Simulating LTE Cellular Systems: an Open Source Framework," *IEEE Trans. on Vehicular Technology*, vol. 60, no. 2, pp.498-513, Feb. 2011.
- [2] J. A. Fernandez, K. Borries, L. Cheng, B. V. K. V. Kumar, D. D. Stancil and F. Bai, "Performance of the 802.11p Physical Layer in Vehicle-to-Vehicle Environments," *IEEE Trans. on Vehicular Technology*, vol. 61, no. 1, pp.3-14, Jan. 2012.
- [3] Q. Wang, P. Fan and K. B. Letaief, "On the Joint V2I and V2V Scheduling for Cooperative VANETs With Network Coding," *IEEE Trans. on Vehicular Technology*, vol. 61, no. 1, pp.62-73, Jan. 2012.
- [4] YouTube, <http://www.youtube.com>.
- [5] C. Xu, G.-M. Muntean, E. Fallon and A. Hanley, "A Balanced Tree-based Strategy for Unstructured Media Distribution in P2P Networks," in *Proc. IEEE ICC*, May 2008.
- [6] C. Xu, G.-M. Muntean, E. Fallon and A. Hanley, "Distributed storage-assisted data-driven overlay network for P2P VoD services," *IEEE Trans. on Broadcasting*, vol. 55, no. 1, pp.1-10, March 2009.
- [7] C.-L. Chang and S.-P. Huang, "The interleaved video frame distribution for P2P-based VoD system with VCR functionality," *Computer Networks*, vol. 56, no.5, pp.1525-1537, March 2012.
- [8] Y. Zhou, T. Fu and D. M. Chiu, "A Unifying Model and Analysis of P2P VoD Replication and Scheduling," in *Proc. IEEE INFOCOM*, 2012.
- [9] C. Lee, E. Hwang and D. Pyeon, "A Popularity-Aware Prefetching Scheme to Support Interactive P2P Streaming," *IEEE Trans. on Consumer Electronics*, vol. 58, no.2, pp. 382-388, May 2012.
- [10] W. Wu and J. C. S. Lui, "Exploring the Optimal Replication Strategy in P2P-VoD Systems: Characterization and Evaluation," in *Proc. IEEE INFOCOM*, 2011.
- [11] W. -P. Ken Yiu, Xing Jin and S. -H. Gary Chan, "VMesh: Distributed Segment Storage for Peer-to-Peer Interactive Video Streaming," *IEEE J. Sel. Areas Commun.*, vol. 25, no.9, pp. 1717-1731, Dec. 2007.
- [12] S. -H. Gary Chan and W. -P. Ken Yiu, "Distributed storage to support user interactivity in peer-to-peer video," *US Patent*, number 7925781, Apr. 2011.
- [13] Y. Hsieh and K. Wang, "Dynamic overlay multicast for live multimedia streaming in urban VANETs," *Computer Networks*, vol. 56, no. 16, pp.3609-3628, Nov. 2012.
- [14] L. Zhou, Y. Zhang, K. Song, W. Jing and A. V. Vasilakos, "Distributed Media Services in P2P-Based Vehicular Networks," *IEEE Trans. on Vehicular Technology*, vol. 60, no. 2, pp. 692-703, Feb. 2011.
- [15] N. N. Qadri, M. Fleury, M. Altaf and M. Ghanbari, "Multi-source video streaming in a wireless vehicular ad hoc network," *IET Communications*, vol. 4, no. 11, pp.1300-1311, July 2010.
- [16] Z. Yang, M. Li and W. Lou, "CodePlay: Live multimedia streaming in VANETs using symbol-level network coding," in *Proc. IEEE ICNP*, 2010.
- [17] I. Stoica, R. Morris, D. L. Nowell, D. R. Karger, M. F. Kaashoek, F. Dabek and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup protocol for internet applications," *IEEE/ACM Trans. Networking*, vol.11, no. 1, pp. 17-32, Feb. 2003.

- [18] C. Canali, M.E. Renda, P. Santi and S. Buresi, "Enabling Efficient Peer-to-Peer Resource Sharing in Wireless Mesh Networks," *IEEE Trans. on Mobile Computing*, vol. 9, no. 3, pp. 333-347, Mar. 2010.
- [19] C.-L. Liu, C.-Y. Wang and H.-Y. Wei, "Mobile Chord: Enhancing P2P Application Performance over Vehicular Ad Hoc Network," in *Proc. IEEE Globecom*, 2008.
- [20] C.-L. Liu, C.-Y. Wang and H.-Y. Wei, "Cross-Layer Mobile Chord P2P Protocol Design for VANET," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 6, no. 3, pp. 150-163, Sept. 2010.
- [21] W. Ding and S. S. Iyengar, "Bootstrapping Chord over MANETs - All Roads Lead to Rome," in *Proc. IEEE WCNC*, 2007.
- [22] A. Forestiero, E. Leonardi, C. Mastroianni, and M. Meo, "Self-chord: a bio-inspired P2P framework for self-organizing distributed systems," *IEEE/ACM Trans. on Networking*, vol. 18, no. 5, pp. 1651-1664, Oct. 2010.
- [23] W. C. Ao, P.-Y. Chen and K.-C. Chen, "Rate-Reliability-Delay Tradeoff of Multipath Transmission Using Network Coding," *IEEE Trans. on Vehicular Technology*, vol. 61, no. 5, pp. 2336-2342, June 2012.
- [24] C. Ahlund, R. Brannstrom and A. Zaslavsky, "Traffic load metrics for multi-homed mobile IP and global connectivity," *Telecommunication Systems*, vol. 33, no. 1-3, pp. 155-185, 2006.
- [25] E. Nordmark and M. Bagnulo, "Shim6: Level 3 Multihoming Shim Protocol for IPv6," *RFC 5533*, June 2009.
- [26] T. Okimoto, D. Wing, D. Miles and S. Matsushima, "IPv6 Multihoming without Network Address Translation," *Internet Draft*, IETF, Feb 2012, <draft-ietf-v6ops-ipv6-multihoming-without-ipv6nat-04>, work in progress.
- [27] A. Ford, C. Raiciu, M. Handley, S. Barre and J. Iyengar, "Architectural Guidelines for Multipath TCP Development," *RFC 6182*, March 2011.
- [28] C. Xu, E. Fallon, Y. Qiao, G.-M. Muntean, X. Li and A. Hanley, "Performance Evaluation of Distributing Real-time Video over Concurrent Multipath," in *Proc. IEEE WCNC*, 2009.
- [29] C. Xu, E. Fallon, Y. Qiao, L. Zhong and G.-M. Muntean, "Performance Evaluation of Multimedia Content Distribution over Multi-Homed Wireless Networks," *IEEE Trans. on Broadcasting*, vol. 57, no. 2, pp. 204-215, Jun 2011.
- [30] C. Xu, T. Liu, J. Guan, H. Zhang and G.-M. Muntean, "CMT-QA: Quality-aware Adaptive Concurrent Multipath Data Transfer in Heterogeneous Wireless Networks," *IEEE Trans. on Mobile Computing*, vol. pp. no.99, Aug. 2012.
- [31] G. Piro, L. A. Grieco, G. Boggia, R. Fortuna and P. Camarda, "Two-Level Downlink Scheduling for Real-Time Multimedia Services in LTE Networks," *IEEE Trans. on Multimedia*, vol. 13, no. 5, pp. 1052-1065, Oct. 2011.
- [32] T. Pitoura, N. Ntarmos and P. Triantafyllou, "Replication, Load Balancing and Efficient Range Query Processing in DHTs," *Advances in Database Technology*, LNCS 3896, pp. 131-148, 2006.
- [33] T. Shuminoski and Toni Janevski, "Cross-Layer Adaptive QoS Provisioning for Next Generation Wireless Networks," *International Journal of Research and Reviews in Next Generation Networks*, vol.1, no.1, pp.7-13, March 2011.
- [34] Q. Zhang, W. Zhu and Y. Q. Zhang, "Resource allocation for multimedia streaming over the internet," *IEEE Trans. on Multimedia*, vol.3, no.3, pp.339-355, 2001.
- [35] N. Baldo, F. Maguolo, M. Miozzo, M. Rossi and M. Zorzi, "Miracle: the multi-interface cross-layer extension of ns2," *EURASIP Journal on Wireless Commun. and Networking*, vol. 2010, no. 26, April 2010.
- [36] <https://telecom.dei.unipd.it/tlcrepos/nsmiracle-dev/trunk>.
- [37] D. Kmjzewicz, G. Hertkorn, P. Wagner and C. Rossel, "Simulation of Urban Mobility (SUMO): An Open-source Traffic Simulation," in *Proc. of the 4th Middle East Symposium on Simulation and Modeling*, 2002.
- [38] F. Bai, N. Sadagopan and A. Helmy, "The IMPORTANT framework for analyzing the Impact of Mobility on Performance Of Routing protocols for Adhoc Networks," *Ad Hoc Networks*, vol.1, no.4, pp.383-403, Nov. 2003.
- [39] IEEE 802.16e-2005, "IEEE Standard for Local and Metropolitan Area Networks-Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems".
- [40] The Network Simulator ns-2 NIST add-on - IEEE 802.16 model (MAC+PHY), Technical Report, National Institute of Standards and Technology, January 2009.
- [41] NS2MiracleWimax, <http://sourceforge.net/projects/ns2miraclewimax/>.
- [42] A. Brampton, A. MacQuire, I. A. Rai, N. J. P. Race, L. Mathy and M. Fry, "Characterising User Interactivity for Sports Video-on-Demand," in *Proc. ACM NOSSDAV*, Urbana-Champaign, USA, Apr. 2007.



Changqiao Xu received his Ph.D. degree from Institute of Software, Chinese Academy of Sciences (ISCAS) in Jan 2009. He was an assistant research fellow in ISCAS from 2002 to 2007, where he was research and development project manager in the area communication networks. During 2007-2009, he worked as a researcher in the Software Research Institute at Athlone Institute of Technology, Ireland. He joined Beijing University of Posts and Telecommunications (BUPT) in Dec. 2009, and was an assistant professor from 2009 to 2011. He is currently an associate professor with the Institute of Network Technology, and associate director of the Next Generation Internet Technology Research Center at BUPT. His research interests include wireless networking, multimedia communications and next generation Internet technology. He is a member of IEEE .



Futao Zhao received his B.S. degree from Hangzhou Dianzi University, China in 2010. He is currently working toward the M.S. degree in the Institute of Network Technology at Beijing University of Posts and Telecommunications. His research interests include wireless sensor network, vehicular network and multimedia streaming.



Jianfen Guan received his Ph.D. degrees in communications and information system from the Beijing Jiaotong University, Beijing, China, in Jan. 2010. He is a lecturer in the Institute of Network Technology at Beijing University of Posts and Telecommunications, Beijing, China. His main research interests focus around mobile IP, mobile multicast and next generation Internet technology.



Hongke Zhang received his Ph.D. degrees in electrical and communication systems from the University of Electronic Science and Technology of China in 1992. From 1992 to 1994, he was a postdoctoral research associate at Beijing Jiaotong University, and in July 1994, he became a professor there. He has published more than 150 research papers in the areas of communications, computer networks, and information theory. He is the author of eight books written in Chinese and the holder of more than 40 patents. He is now the Head of the Institute of Network Technology at Beijing University of Posts and Telecommunications, and the chief scientist of a National Basic Research Program of China.



Gabriel-Miro Muntean received the Ph.D. degree from Dublin City University, Ireland for research in the area of quality-oriented adaptive multimedia streaming in 2003. He is a Lecturer with the School of Electronic Engineering at Dublin City University, Ireland. Dr. Muntean is Co-Director of the DCU Performance Engineering Laboratory, Director of the Network Innovations Centre, part of the Rince Institute Ireland and Consultant Professor with Beijing University of Posts and Telecommunications (BUPT), China. His research interests include quality-

oriented and performance-related issues of adaptive multimedia delivery, performance of wired and wireless communications, energy-aware networking and personalised e-learning. Dr. Muntean has published over 120 papers in prestigious international journals and conferences, has authored a book and eleven book chapters and has edited four other books. Dr. Muntean is Associate Editor of the IEEE Transactions on Broadcasting, Associate Editor of the IEEE Communications Surveys and Tutorials and reviewer for other important international journals, conferences and funding agencies. He is a member of ACM, IEEE, and IEEE Broadcast Technology Society.