GrIMS: Green Information-centric Multimedia Streaming Framework in Vehicular Ad Hoc Networks

GrIMS: Green Information-centric Multimedia Streaming Framework in Vehicular Ad Hoc Networks

Changqiao Xu, *Senior Member, IEEE*, Wei Quan, *Member, IEEE*, Hongke Zhang, *Senior Member, IEEE* and Luigi Alfredo Grieco, *Senior Member, IEEE*

***Abstract*—Information Centric Networking (ICN), as a novel network paradigm, is expected to natively support mobility, multicast, and multihoming in Vehicular Ad hoc Networks (VANETs). In this paper, the adoption of ICN principles for multimedia streaming in multihomed VANETs is investigated, with a major emphasis on the tradeoff between Quality of Experience (QoE) and energy efficiency (EnE). To formalize this problem, a cost optimization model is first proposed, based on queueing theory arguments. Then, a novel Green Information-centric Multimedia Streaming framework (GrIMS) is designed to drive the system towards optimal working points in practical settings. GrIMS consists of three enhanced mechanisms for on-demand cloud-based processing, adaptive multi-path transmission and cooperative in-network caching. Finally, a massive simulation campaign has been carried out, demonstrating that, thanks to its core components, GrIMS enables flexible multimedia service provisioning and achieves an improved performance in terms of start-up delay, playbacks continuity and energy efficiency with respect to state-of-the-art solutions.**

***Index Terms*—Information-Centric Networking, Multimedia Streaming, Energy Efficiency, QoE, Vehicular Ad Hoc Networks.**

I. Introduction

In a Vehicular Ad hoc NETwork (VANET), due to high mobility of vehicles, links suffer time-varying and location-dependent intermittence. This issue greatly hinders the provisioning of high quality multimedia services to passengers and drivers [1]-[3]. The current host-centric Internet design requires persistent connectivity between couple of hosts, located several hops away, which is not that easy to achieve even in moderate mobility conditions [4]. Information Centric Networking (ICN) , an emerging paradigm for the future Internet, is expected to provide remarkable benefits to the quality of multimedia streaming in mobile scenarios [5][6]. The key idea of ICN is to replace the traditional *host-based* networking primitives with novel *name-based* ones. Accordingly, a content requester needs not to maintain the permanent connection with a certain host (that acts as a content provider) because all networking operations are driven by content names without any reference to host locators. In this way, ICN becomes able to natively support mobility, nomadic communications, multicast, multi-path routing, and VANETs [7][8].

Recently, ICN-based multimedia services in VANETs are being actively investigated by the research community [9]-[15]. The valuable efforts proposed so far solved several issues related to ICN-based streaming in VANETs (see Sec. II for a deeper analysis of related work). Unfortunately, the road ahead towards fully integrated multimedia systems in VANETs requires further research because of the growing demand for multimedia with an increased resolution (including popular 3D movies, high-definition television (HDTV), and 3D games).

In addition to quality of experience (QoE) challenges, it is worth to note that a low energy consumption is a big concern for video sharing in VANETs [16]. The main energy cost mainly reflects in the following aspects: *(i)* wireless network infrastructure consumes large amounts of energy to guarantee the quality of service (QoS); *(ii)* high probability of packet loss, reconnection, and retransmission exacerbate energy consumptions of mobile vehicles; *(iii)* various multimedia applications with high data rate, *e.g.*, mobile video on-demand, necessitates more energy per bit for a given bit error rate (BER) [17]. Although future electric vehicles (EVs) have large enough batteries to run the whole vehicle system, battery charging is still a well-known technical obstacle. Therefore, how to save energy is still a key concern for EV drivers.

To the best of authors' knowledge, little specific ICN-based research has looked into energy-efficient multimedia delivery in vehicular networks so far. To bridge these gaps, a *green* information-centric framework for *high-quality* multimedia streaming in wireless vehicular networks is investigated hereby, which targets an optimal tradeoff between QoE and energy efficiency (EnE). In summary, the major contributions of this work can be summarized as follows:

- A novel green information-centric multimedia streaming framework (GrIMS) is conceived for heterogeneous VANETs, which exist multiple communication means concurrently. Particularly, all network elements are equipped with information centric engines, thus resulting

This work was supported by the National Basic Research Program of China (973) under Grant 2013CB329102, by the National Natural Science Foundation of China (NSFC) under Grant No. 61522103, 61602030, 61372112 and 61232017, by the Beijing Natural Science Foundation under Grant 4142037.

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

W. Quan and H. Zhang are with the National Engineering Laboratory for Next Generation Internet Technologies, School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China (e-mail: weiquan@bjtu.edu.cn, hkzhang@bjtu.edu.cn).

L. A. Grieco is with the Department of Electrical and Information Engineering, Politecnico di Bari, Bari, Italy (e-mail: a.grieco@poliba.it).

in a distributed Information Centric Cloud (IC-Cloud). When a vehicle asks for a content name, IC-Cloud determines how to deliver the corresponding data based on the requested contents, which enables to schedule available resources adaptively to guarantee QoE and EnE.
- System models are proposed to characterize the multimedia retrieval in GrIMS. Specifically, a queuing model with differentiated service rates is built for the simplified GrIMS system, where two communication technologies have been considered, namely vehicle to vehicle/roadside unit (V2V/R) using WAVE/802.11p and vehicle to infrastructure (V2I) using 4G/LTE. The average sojourn time of content requests is derived in explicit form based on the queuing model. Furthermore, a joint cost optimal problem is formulated to maximize the QoE of multimedia signals while minimizing energy consumptions.
- Three heuristic mechanisms are further designed in GrIMS framework to enable VANETs optimization in practical settings: (*i*) on-demand cloud-based processing; (*ii*) adaptive multi-path transmission; and (*iii*) cooperative in-network caching.
- A powerful simulation platform for GrIMS based on NS-3 has been developed and an extensive simulation campaign has been carried out. Its outcomes show that GrIMS outperforms other two state-of-the-art solutions VMesh [52] and QUVoD [23] in terms of QoS and QoE while pursuing energy saving objectives.

II. Related Works

This section summarizes the most significant scientific contributions related to both traditional host-centric and emerging information-centric content delivery in VANETs, as well as ICN-based energy-efficient solutions. Table I summarizes the comparisons of current related works.

Traditional host-centric content delivery. In this field, most of proposed approaches leverage optimization frameworks and P2P technologies [18]. F. Malandrino *et al.* modeled the content downloading process as an optimization problem, and maximize the overall system throughput [19]. O. Cruces *et al.* devised solutions for the selection of carriers and data chunks at the APs, observing that carry-and-forward transfer can greatly increase the download rate of vehicular users [20]. L. Zhou *et al.* developed fully dynamic service schema that aims to maximize the total user-satisfaction and achieve a certain amount of fairness in P2P-based vehicular networks [21]. N. Qadri *et al.* studied P2P multimedia exchange in vehicular networks and propose a novel slice compensation scheme employing spatial Multiple Description Coding to provide error resilience [22]. C. Xu *et al.* proposed an adaptive QoE driven user-centric video on-demand (VoD) scheme in urban multi-homed P2P-based vehicular networks [23]. *These approaches, albeit powerful, inherit the drawbacks of host-centric networks, such as limited support to mobility and multicast communications.*

Information-centric data delivery. With ICN, it is possible to overcome the limited support to mobility in IP networks. Amadeo *et al.* [10] confirmed the potential of ICN as a promising solution for future vehicular networks. G. Arnould *et al.* [11] first extended and modified the content-centric networking (CCN) (a special instance of ICN) for hybrid VANETs and implemented a new type of unsolicited packet to allow vehicular networks to produce emergency messages. P. TalebiFard *et al.* [12] adopted a selective random network coding approach and proposed a content-centric solution for information dissemination in vehicular network environments. Additionally, L. Wang *et al.* applied the Named Data Networking (NDN) concept in V2V communications, and proposed a data name design to develop a simple traffic information dissemination application [13]. *It is worth to remark that all these proposals focus on general data retrieval, and do not consider the specific multimedia context, which, as well known, poses stronger QoS requirements on throughput and delay.*

TABLE I
Comparison of Existing Works

Category	Green	ICN	Cloud	VANET	Multimedia
P2P optimization [18]-[23]	×	×	√	√	√
ICN data delivery [10]-[13]	×	√	×	√	×
ICN multimedia [14],[15]	×	√	×	×	√
Energy optimization [16],[17],[24]	√	×	√	×	×
ICN energy saving [25]-[30]	√	√	×	×	√
GrIMS proposed in this paper	√	√	√	√	√

Information-centric multimedia delivery. Recent studies are proposing the ICN paradigm also in support to multimedia services. A. Detti *et al.* presented a cooperative multimedia streaming application running on top of ICN as proof of concept [14]. Additionally, Z. Li *et al.* introduced a cooperative caching strategy for the treatment of large video streams with on-demand access [15]. Unfortunately, all these proposals still focus on multimedia delivery over static ICN networks and do not consider ICN-based multimedia delivery in VANETs.

ICN-based energy-efficient solution. Researchers have proposed different approaches to save energy, including green-cloud, greenradio, energy-aware adaption [24]. Recently, green communications via ICN is attracting more and more attention of researchers [25]. K. Pentikousis outlined open research issues in the design of energy-efficient future mobile Internet [26]. Piro *et al.* integrated some promising future Internet paradigms (including ICN) to provide effective and efficient solutions for mobile cloud [27]. Lee *et al.* proposed a green Internet solution using CCN and stated that CCN is more energy efficient than conventional CDNs and P2P solutions [28]. Butt *et al.* conducted an energy consumption analysis of ICN compared to IP-based network in a video streaming scenario [29]. Choi *et al.* investigated the minimum energy consumption of ICN with optimal cache locations [30]. *These first research outcomes, albeit precious, requires further efforts to embrace green multimedia services in VANETs too.*

In order to complement the precious efforts and move a further step towards a higher degree of knowledge in this articulated research field, this paper (for the first time, to the best of authors' knowledge) proposes an integrated energy-efficient multimedia streaming framework in VANETs by leveraging novel information-centric features.

III. Overview of GrIMS Architecture

Named Data Networking (NDN) [32] is a specific ICN flavor, leveraged in GrIMS[1]. It is based on two kinds of messages: *Interest* and *Data*. The former are used to ask a given content. The latter encapsulate the contents transmitted across the network. All the nodes are networked using the NDN forwarding engine, which includes the NDN-based forwarding strategies. To leverage the potential of ICN, GrIMS is equipped with an improved NDN forwarding engine.

Fig. 1 illustrates the architecture and work flow of GrIMS. In a nutshell, the basic delivery behavior can be summarized as three steps: *i*) a requester generates and transmits an *Interest* packet, which contains the name of the content he is willing to download or obtain; *ii*) NDN nodes process and forward the *Interest* based on the carried content name till it reaches a node in possess of the requested content; *iii*) a *Data* packet, encapsulating the requested content, is created and sent back to the requester. It is worth to note that the requester has not any knowledge about potential content providers and delivery path when it emits the *Interest* packet. It is the ICN network that is in charge of querying and resolving the *Interest*, searching available content providers, and deciding how forward it according to the in-network operations. With this feature, video consumers just issue specific request messages for named chunks, by leaving the network to route their queries towards the appropriate video providers.

To be specific, GrIMS is split into two logical tiers: *computing/control tier* and *forwarding/caching tier*. The former is mainly in charge of content searching, resource scheduling, and delivery decision. The latter manages the data forwarding and content caching in each node. In the computing/control tier, all nodes in GrIMS serves as an information centric cloud [27], which includes not only by stationary data centers and network infrastructure but also by mobile vehicles. This distributed IC-Cloud allows vehicle nodes to: (*i*) join or leave the cloud to provide content delivery adaptively; (*ii*) resolve the queries and control the content-based routing, (*iii*) actively cooperate to cache contents. In the forwarding/caching tier, caching in each node makes GrIMS as a huge resource pool, providing a vehicular cloud networking (VCN) paradigm [35].

Bases on this design, GrIMS is able to greatly relieve the effect from transmission interruption due to node's mobility. Besides, thanks to the IC-Cloud, GrIMS can easily solve the source mobility problem by routing interests directly to the mobile sources [36]. The detailed workflow of GrIMS are summarized as follows:

Step (1): When one requester in the vehicle V_i desires a multimedia content, an *Interest* request will be sent that asks for that specific content.

Step (2): Any node of the IC-Cloud, such as V_j, once received the *Interest* request, will trigger a cloud-based process in the computing/control tier. On one hand, the neighbor nodes of V_j in IC-Cloud start content searching in their local forwarding engine to find the nearby providers. On the other hand, the neighbor nodes continuously notify their status to

[1]The GrIMS formulation does not preclude the adoption of other ICN architectures different from NDN.

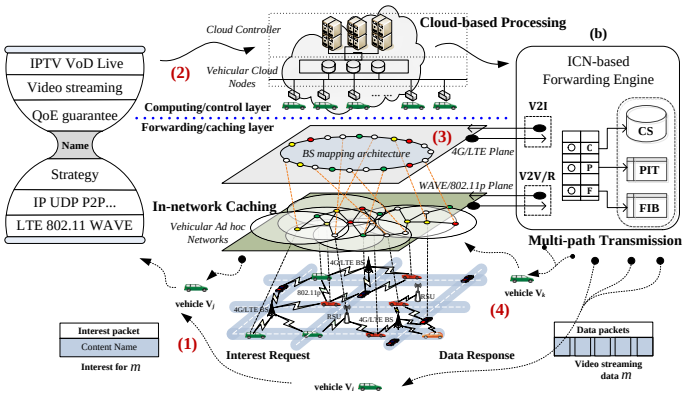


Fig. 1. GrIMS: green information-centric multimedia streaming architecture.

the cloud controller, including content list, computing power, available bandwidth, storage space and so on. To reduce the signalling costs, we can borrow the *Interest* messages to carry these signals. With these knowledge, the IC-Cloud will compute and schedule appropriate cloud nodes globally to pursue a content delivery decision (*detailed mechanisms will be introduced in the following sections*), which is more efficient than a pure local search based on nearby nodes. Through this process, the IC-Cloud will find the proper content providers, determine the delivery path, and notify related nodes to forward or cache the desired data.

Step (3): After that, the work flow moves to the forwarding/caching tier. Here, multiple logical communication planes are embraced by GrIMS to enable multihoming (*i.e.,* one plane for every communication technology). Each node can forward the desired data by using of different service faces. For example, two mainstream technologies are considered in the following experiments: 4G/LTE for V2I and WAVE/802.11p for V2V/R. The WAVE/802.11p plane can provide short-distance communication with a smaller energy consumption, while the 4G/LTE plane can provide long-distance communication with a bigger energy consumption. Each network node can also simultaneously use multiple available service faces, which are supported natively in the NDN forwarding engine.

Step (4): Finally, the content will be delivered to the content requester via one or multiple paths. Furthermore, some of in-path caching nodes can decide to cache the contents in local content store to make themselves potential providers later on.

GrIMS builds a complete framework with many unique characteristics. It is able to undertake high workloads and support fast processing by means of distributed cloud operations. Although offloading some computing work to the cloud can result in additional communication overhead, GrIMS can bring great benefits and resolve some challenging issues, thus, guaranteeing the high real-time quality for multimedia streaming in such a dynamic VANET scenario. Besides, GrIMS allows the network to allocate/cache resources dynamically according to the requirements of content requests and adapt to the harsh wireless conditions of VANETs.

To further these main features in GrIMS clearly, we elaborate the system models leveraging the differentiated queuing model and the multi-objective optimal model, which are detailed in the next section.

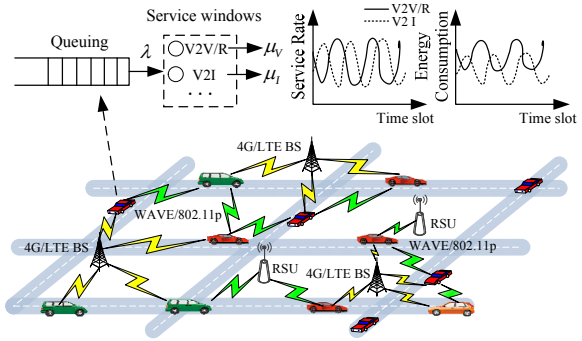


Fig. 2. System model: (a) urban wireless vehicular network (b) queuing model with differentiated service windows

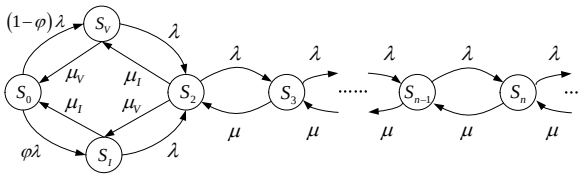


Fig. 3. The state-transition-rate diagram of our model

IV. System Models

We consider a simple urban infrastructure-based vehicular network as shown in Fig. 2. Each vehicle is equipped with two different wireless communication interfaces[1]. Due to different PHY and MAC protocols adopted, these technologies reflect to different transmission rates and energy efficiencies. To model this differentiated and articulated service process, a queueing system with multiple windows is formulated hereby. In particular for the scenario in Fig. 2, a simple queuing system $\mathcal{Q}$ can be adopted with two different service windows $w^{(V)}$ and $w^{(I)}$. With such a queuing system, all requests are queued to be processed in a big enough buffer, and each of them may be served by either $w^{(V)}$ or $w^{(I)}$.

To ease the description, Table II lists key notations in the model. In detail, the multimedia content request process is structured in two levels, *content* and *chunk*[2]. The chunk request arrival process is modeled as a Markov Modulated Rate Process (MMRP) [31], where requests for contents follow a Poisson process of intensity λ_0. Furthermore, assume that the content length expressed in number of chunks is a geometrically distributed random variable N_c with mean σ [34], which is as follows:

$$\mathrm{P}\left(N_c = s\right) = \frac{1}{\sigma}\left(1 - \frac{1}{\sigma}\right)^{s-1}, s = 1, 2, 3 \cdots \tag{1}$$

Let μ denote the total service rate (*i.e.* bandwidth) of service windows, $\lambda = \lambda_0 \cdot \sigma$ be the intensity of chunk requests and μ_V, μ_I denote the mean service rates in two windows respectively. Since two windows can work concurrently, then it has $\mu = \mu_V + \mu_I$.

For this queuing system, the queue discipline is described as follows. When both windows are idle, one multimedia chunk request is served by $w^{(I)}$ and $w^{(V)}$ with the probability of φ and $1-\varphi$ respectively ($0 \le \varphi \le 1$). When $w^{(V)}$ is busy and $w^{(I)}$ is idle, $w^{(I)}$ starts serving a new request. All waiting requests which arrived after this request move forward one position. In a similar way, when $w^{(I)}$ is busy and $w^{(V)}$ is idle, $w^{(V)}$ starts serving a new request. Based on this, the state-transition-rate diagram of this model is shown in Fig. 3. The status $\mathcal{S}_0$ denotes no request in the system, two service windows are idle; the status $\mathcal{S}_V$ denotes one request is serving by $w^{(V)}$ in the system, the $w^{(I)}$ is idle; the status $\mathcal{S}_I$ denotes one request is serving by the window $w^{(I)}$ in the system, the $w^{(V)}$ is idle; the status of $\mathcal{S}_2$ denotes two requests are serving by two windows; the status of $\mathcal{S}_3, \mathcal{S}_4, ..., \mathcal{S}_n$ denotes $3, 4, ...n$ requests exist in the queuing system, two requests are being served, other ones are waiting to be served.

We use the mean sojourn time in the queuing system to determine the chunk delivery time, which affects the quality of multimedia delivery. In the following, the mean sojourn time of each request in the system will be derived.

Let p_k denote the steady-state probability of the status $\mathcal{S}_k$ in the queuing system. In particular, p_V and p_I represent the probability of the status $\mathcal{S}_V$ and $\mathcal{S}_I$ respectively. Clearly, it has $p_k \ge 0, k \in \{0, V, I, 2, \cdots, n\}$.

According to the Kolmogorov equations, the following set of $n+2$ equations are valid in our queuing system,

$$\left.\begin{aligned}
\lambda p_0 &= \mu_V p_V + \mu_I p_I \quad (2.1)\\
(\lambda + \mu_V) p_V &= \mu_I p_2 + (1-\varphi)\lambda p_0 \quad (2.2)\\
(\lambda + \mu_I) p_I &= \mu_V p_2 + \varphi \lambda p_0 \quad (2.3)\\
(\lambda + \mu) p_2 &= \lambda p_V + \lambda p_I + \mu p_3 \quad (1.4)\\
&\vdots\\
(\lambda + \mu) p_n &= \mu p_{n+1} + \lambda p_{n-1} \quad (2.n+2)
\end{aligned}\right\} \tag{2}$$

Let $\rho = \frac{\lambda}{\mu}$, and $\alpha = \frac{\mu_V}{\mu_I}$. The expressions of stationary-

TABLE II
Key Notations

Notations	Definition
$\mathcal{Q}$	a queuing system
λ_0, λ	arrival intensity of content, chunk requests
σ	average number of chunks in one content
$w^{(V)}$, $w^{(I)}$	service windows by means of V2V/R, V2I
μ_V, μ_I, μ	average service rate in $w^{(V)}$, $w^{(I)}$, total service rate
φ	probability of serving by $w^{(I)}$
p_n	steady-state probability of the status $\mathcal{S}_n$
$\mathcal{W}_s$	mean sojourn time in the queuing system $\mathcal{Q}$
m_s	average number of tasks in the queuing system
c_e, c_V, c_I	energy cost per time unit, ones using $w^{(V)}$ or $w^{(I)}$
ρ	ratio of λ/μ
α	ratio of μ_V/μ_I
c_q	QoE loss by waiting in a time unit
t	the observation period of time
$\mathcal{C}_q(\cdot)$	QoE loss cost in the queuing system
$\mathcal{C}_e(\cdot)$	energy cost in the queuing system
$\mathcal{U}(\cdot)$	total cost in the queuing system
x	number of autoregressive terms
z	number of nonseasonal differences
q	number of lagged forecast errors
$\mathcal{P}_n$, Y_n	number and difference of required computing power
b	playback rate
Σ_0	size of buffered content in the time of τ_0
u_x	available upload rate by x delivery means
$\mathcal{L}_c$, $\Delta\mathcal{L}_c$	load capacity and available load capacity of one node
$\mathcal{S}_c$, $\bar{\mathcal{S}}_c$, $\mathcal{S}_c{}'$	storage capacity, total ones and used ones
r_c, $\bar{r}_c$, $r_c{}'$	transmission capacity, total ones and used ones
τ_l	the longest tolerable waiting time
$\mathcal{R}, \mathcal{R}_i$	number of requests, number of requests for M_i
$\mathcal{N}_p, \mathcal{N}_{cc}$	number of providers and cooperative caching nodes

[1]The same approach can be extended to embrace more than two interfaces and technologies other than 4G and 802.11p.

[2]An entire multimedia content is made of a sequence of chunks.

state probabilities for our queuing system are (the derivation is detailed in Appendix A):

$$p_k = \begin{cases} \frac{\rho}{1+2\,\rho}\,(1+\alpha)\,(\rho+\varphi)\,p_0, & k \text{ is } V \\ \frac{\rho}{1+2\,\rho}\,\frac{1+\alpha}{\alpha}\,(\rho+1-\varphi)\,p_0, & k \text{ is } I \\ \frac{\rho^k}{1+2\,\rho}\,\frac{1+\alpha}{\alpha}\,[1+(1+\alpha)\,\rho-(1-\alpha)\varphi]\,p_0, & k \notin \{0,V,I\} \end{cases} \tag{3}$$

According to the normalizing condition:

$$p_0 + p_V + p_I + \sum_{k=2}^{\infty} p_k = 1 \tag{4}$$

The value of p_0 can be determined:

$$p_0 = \frac{1-\rho}{1+\rho\,[1+(1+\alpha^2)\,\rho-(1-\alpha^2)\,\varphi]/\alpha\,(1+2\,\rho)} \tag{5}$$

Then, it yields the average number of tasks m_s in the system as follows (the derivation is detailed in Appendix B):

$$\begin{aligned} m_s &= p_V + p_I + \sum_{n=2}^{\infty} n p_n \\ &= \tfrac{\rho(1+\alpha)}{1-\rho} \cdot \tfrac{1+(1+\alpha)\,\rho-(1-\alpha)\varphi}{\alpha(1+2\,\rho)+\rho[1+(1+\alpha^2)\,\rho-(1-\alpha^2)\varphi]} \end{aligned} \tag{6}$$

Finally, the average sojourn time in the system for each request task can be obtained as

$$\begin{aligned} \mathcal{W}_s &= \tfrac{m_s}{\lambda} \\ &= \tfrac{(1+\alpha)}{\mu(1-\rho)} \cdot \tfrac{1+(1+\alpha)\rho-(1-\alpha)\varphi}{\alpha(1+2\rho)+\rho[1+(1+\alpha^2)\rho-(1-\alpha^2)\varphi]} \end{aligned} \tag{7}$$

This simplified model is able to catch the main key performance indicator of a single queuing unit in the system.

Based on the above model, we further study its optimization problem, which focuses on two kinds of costs that account for energy consumption and QoE, respectively. In particular, the cost brought by QoE loss is positively correlated to the sojourn time $\mathcal{W}_s$ in the system, which depends on λ, μ_V, μ_I and φ according to the Eq. (7). Based on this, a simple assumption is proposed as follows:

Assumption 1: The cost brought by the QoE loss of multimedia delivery is related to the waiting time in the queue, as follows:

$$\mathcal{C}_q = \mathcal{W}_s \cdot c_q \tag{8}$$

where c_q is the average QoE loss brought by a time delay.

The cost due to energy consumption may be affected by many factors. We focus on the following ones: (1) the sojourn time in the service queue reflects the working time for all service windows of the queue. A higher sojourn time will result in a higher energy cost; (2) to increase the service rate, the service window requires a higher power, which will bring a higher energy consumption; (3) assuming that the service rate of each service window is big enough, a higher arrival rate λ means more tasks required to be served and accomplished, which will also result in a higher energy cost; (4) energy cost can be also adjusted by changing the serving probability φ of window $w^{(I)}$. Based on those considerations and by letting c_I and c_V denote the energy costs associated to two different service windows $w^{(I)}$ or $w^{(V)}$, respectively. Based on the above considerations, we propose another foundational assumption for the energy cost as follows:

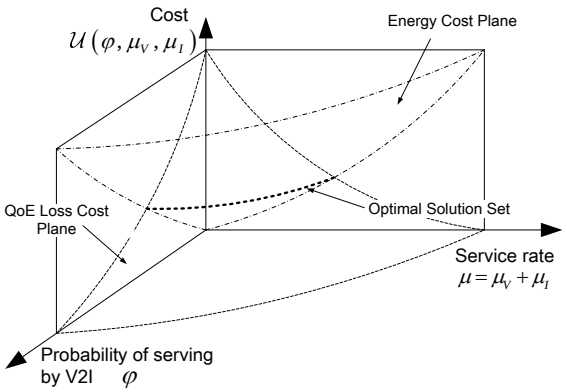


Fig. 4. Cost optimization in the queuing system with regard to φ and μ

Assumption 2: The cost brought by energy consumption can be simplified as follows:

$$\mathcal{C}_e = [\varphi\lambda c_I + (1-\varphi)\,\lambda c_V]\,\mathcal{W}_s \tag{9}$$

Combining the Eq. (8) and Eq. (9), the total cost $\mathcal{U}$ is derived as follows:

$$\begin{aligned} \mathcal{U}(\varphi, \mu_V, \mu_I) &= \beta_q \cdot \mathcal{C}_q + \beta_e \cdot \mathcal{C}_e \\ &= \mathcal{W}_s\,[\beta_q c_q + \beta_e \varphi \lambda c_I + \beta_e (1-\varphi)\,\lambda c_V] \end{aligned} \tag{10}$$

where β_q, β_e are the weighting factors for QoE loss cost and energy cost, respectively.

The minimization of the total cost is the objective of our optimization problem. For different values of φ, μ_V, μ_I, there are different values of $\mathcal{U}$. Accordingly, an optimal solution set can be found to minimize the total cost $\mathcal{U}(\varphi^*, \mu_V^*, \mu_I^*)$ by a greedy algorithm.

According to Eq. (10), Fig. 4 shows the two cost planes in terms of energy and QoE loss with a constant request arrival rate, and illustrates how the cost varies with the service rate μ and the probability φ in different delivery approaches. This plot highlights that the different variables of the system are highly coupled to each other. Moreover, in a real VANET it is hard to solve the afore-stated optimization problem in a centralized and globally valid way because the variables that feed the problem are time-varying. In other terms, the mobility of nodes make almost useless any attempt to solve the optimization problem with a greedy approach because once the solution is found the VANET conditions are already changed and the solution cannot be applied to the system.

Based on above analysis, a set of heuristic algorithms are required to strengthen this green information-centric multimedia delivery framework, and make it more practical.

V. GrIMS Mechanisms and Algorithms

This section focuses on the efficient mechanisms of GrIMS in three aspects: cloud-based processing, multi-path transmission and in-networking caching.

GrIMS integrates three main functions into the ICN-based efficient multimedia delivery: *cloud-based processing, multi-path transmission and in-network caching*.

Algorithm 1: On-demand Cloud-based Processing Mechanism

while a number of requests occur in IC-Cloud **do**
  for each time period t_k in $\{t_0, t_1, \cdots, t_n\}$ **do**
    get the number of requests $\mathcal{I}_k$;
    calculate the computing power $\mathcal{P}_k$ by Eq. (11)
  end for
  build the time series of computing power demand $\{\mathcal{P}_n\}$;
  set $z = 1$, $Y_n = \mathcal{P}_n$;
  while $\{Y_n\}$ is not stationary **do**
    $Y_n = d^z \mathcal{P}_n \ (n > z)$;
    $z \leftarrow z + 1$;
  end while
  derive the *ARMA(p,q)* of $\{Y_n\}$;
  use the Gram-Schmidt orthonormalization procedure to determine the parameters *p*, *q* of *ARMA(p,q)*;
  derive the *ARIMA(p,z,q)*;
  predict the $\mathcal{P}_{n+1}$ according to *ARIMA(p,z,q)*;
  $\Delta\mathcal{P} \leftarrow \mathcal{P}_{n+1} - \mathcal{P}_n$;
  send out the *Interest* of "wake-on-demand" carried with $\Delta\mathcal{P}$;
  each node received this *Interest* adjusts its working status;
end while

A. On-demand cloud-based processing

As mentioned before, triggered by the *Interest* request, IC-Cloud is in charge of computing and managing how to deliver the desired content to the requester. Clearly, the workload contributed by this operation varies according to the aggregate rate of content requests. In these conditions, on demand adaptation of computing resources is required to pursue energy efficiency while fulfilling QoE guarantees.

In GrIMS, it is assumed that the requirement of computing power $\mathcal{P}$, *e.g.*, processing rate or memory space, is proportional to the number of *Interest* requests $\mathcal{N}$ at a certain time, which can be formalized as follows:

$$\mathcal{P} = \mathcal{N} \cdot \varepsilon \tag{11}$$

where ε is the computing power unit for each request *Interest* which can be estimated by historical statistic data.

To effectively allocate available resources, an accurate forecast of computing power is required. In fact, some time is required to execute the adaptation of resources in a real cloud. For example, the Amazon EC2 may spend at least several minutes to increase or decrease the computing capacity [37]. To estimate the required computing power, the time $\mathcal{T}$ is divided into $\mathcal{K}$ consecutive and disjoint time intervals: $\{\tau_1, \tau_2, \cdots, \tau_n, \cdots, \tau_k\}$. Moreover, an Autoregressive Integrated Moving Average (ARIMA) scheme [38] is adopted in GrIMS to allocate computing resource adaptively. ARIMA has been proven to be suitable for the cases where data show evidence of non-stationarity, where an initial differencing step can be applied to reduce the non-stationarity. In our system, the computing power is highly dependent on task amounts, which show the non-stationarity. Therefore, it is feasible to use ARIMA for computing power forecast.

Algorithm 1 illustrates the on-demand cloud-based processing mechanism in GrIMS, which is also detailed in the following. First, the IC-Cloud records the *Interest* request status in each cycle time τ_n and calculates the computing resource according to Eq. (11). Then, it can form a stochastic time series $\{\mathcal{P}_n\}$, where $\mathcal{P}_n$ is the number of required computing power in cloud in n-th period time. Based on this, the $ARIMA(x, z, q)$ model for $\{\mathcal{P}_n\}$ can be built to predict the $\mathcal{P}_{n+1}$, where x is the number of autoregressive terms, z is the number of nonseasonal differences, and q is the number of lagged forecast errors in the prediction equation. To determine the values of these parameters, $\mathcal{P}_n$ is first processed as a stationary series, which is suitable for the Autoregressive Moving Average (ARMA) model $ARMA(x, q)$. This process can be finished by *recursive difference calculation* until it follows the conditions of stationary series, which is shown as follows:

GrIMS://resource/computing/wake_on_demand/	:1	:10 processors|2014020513-14|

GrIMS://resource/computing/wake_on_demand/	:0	:-5 processors|2014020513-14|

Fig. 5. The samples of computing resource "waking-on-demand" Interest

$$Y_n = d^z \left(\mathcal{P}_n\right), (n > z) \tag{12}$$

where $d^z(\cdot)$ is the z order difference operator.

Then, a stationary series $\{Y_n\}$ can be obtained, which is used to build the $ARMA(x, q)$. The values of x, q can be determined according to the Gram-Schmidt orthonormalization procedure [39]. With the knowledge of x and q, the values of Y_{n+1} and $\mathcal{P}_{n+1}$ can be predicted, which denote the required computing power. Then, the controller node of IC-Cloud adjusts the number of computing nodes by means of a novel kind of *Interest*, which is used to search the *available computing resource* and *redundant computing resource* in GrIMS. The format of this "waking-on-demand" *Interest* is shown in Fig. 5. In particular, the segment identifier can be set to "1" for adding available computing resource and "0" for removing redundant computing resource. The attribute labels indicate the amount of computing power to be increased or decreased and the time period. Based on this, each node receiving this *Interest* will execute a fast name lookup [40][41] and decide how to adjust its work status. For example, it may wake up the available computing module in case the identifier is equal to "1" or transfer itself to the inactive status, viceversa. Each time the *Interest* is forwarded to one node, the attribute label is updated to reflect the missing adaptation operations until the number is equal to zero (*i.e.* "10 processors" is modified as "8 processors" when 2 processor are turned off). According to this process, the required computing power in the cloud can be allocated dynamically, which can reduce unnecessary energy consumption.

B. Adaptive multi-path selection

This section focuses on how to efficiently utilize multiple delivery paths, which depends on the parameter φ in the proposed model. For ease of description, it is assumed that the energy cost incurred by V2V/R communications is smaller than V2I ones. This assumption is valid because the energy consumption strongly relies on the transmission power and distance during the wireless communications. The high density of vehicles and RSUs in urban areas means that the communication distance is short when using V2V/R. Therefore, to reduce the energy consumption, the V2V/R interface is considered first, and then the V2I one.

Algorithm 2: Adaptive Multi-path Transmission Mechanism

while a node n_k requests for video segment M_i **do**
  get the information of b, Σ_0 and t;
  estimate the available upload rate $u_{V/R}$ by V2V/R;
  estimate the available upload rate u_I by V2I;
  if $u_{V/R} \geq b - \Sigma_0/t$ **then**
    deliver M_i to n_k with the *V2V/R* path;
  else if $u_I \geq b - \Sigma_0/t$ **then**
    deliver M_i to n_k with the *V2I* path;
  else
    deliver M_i to n_k with both *V2V/R* and *V2I* paths;
    enhanced caching is conducted by Algorithm 3;
  end if
end while

On the other hand, the QoE of multimedia streaming has to be guaranteed as a precondition. In this paper, the QoE contraint will be encoded with the continuity of the stream.

Condition 1: In order to guarantee the QoE of multimedia streaming, playout interruptions should be avoided as much as possible. Accordingly, over a sufficiently small time interval t, the following condition should hold:

$$b \cdot t \leq \Sigma_0 + u_x \cdot t \tag{13}$$

where b denotes the playback rate, Σ_0 denotes the size of buffered content in the time of τ_0, u_x denotes the available upload rate by x delivery means. This condition presents a practical method to determine the parameter ϕ in our system model.

According to formula (13), to smooth video playback in t, the upload rate should satisfy the follow condition:

$$u_x \geq b - \Sigma_0/t \tag{14}$$

Algorithm 2 presents the rule of minimizing energy cost with a guaranteed QoE level, which also provides a special method to decide the parameter φ in our proposed model. The delivery paths are determined in the following steps. Firstly, when a node in IC-Cloud receives a request for a video segment, the IC-Cloud starts to collect the information of playback rate, buffered content size, and also can estimate its available upload rate in different transmission interfaces. The estimation method we used is based on the work [23]. If the V2V/R interface provides an available upload rate which satisfies the formula (14), the V2V/R interface with WAVE/802.11p is first selected for multimedia delivery. If not, the IC-Cloud estimates the upload rate from associated BS (assuming one node only connects one BS), if the condition is satisfied, then V2I interface with 4G/LTE is adopted. In particular, if previous two still cannot satisfy the requirement, both paths can be adopted to guarantee the delivery multimedia streaming and a cooperative caching is triggered (detailed in the next section). When multiple different transfer paths are used [42], network coding technologies [43] can be adopted to reorganize the data flow to avoid the disorder of packets.

C. Cooperative in-network caching

This section focuses on the management of available content providers, hence, adjusting the service process rate μ in the proposed model. In GrIMS, a cooperative caching strategy is executed in order to increase the number of actual providers

Algorithm 3: Cooperative In-network Caching Mechanism

while a large number of requests $\mathcal{N}_i$ for video segment M_i **do**
  IC-Cloud gets $\Delta\mathcal{L}_c$ of nodes nearby the requester by Eq. (16);
  get the required number of caching nodes $\mathcal{N}_{cc}$ by Eq. (19);
  sort and select $\mathcal{N}_{cc}$ cooperative nodes with biggest $\Delta\mathcal{L}_c$ as Ω;
  for each cooperative node n_k in Ω **do**
    if video segment M_i passes n_k **then**
      n_k caches the video segment M_i;
    end if
    n_k requests the following j video segments M_{i+j};
    n_k caches video segments M_{i+j};
  end for
end while

(and hence μ_V and μ_I) subject to the constraints on the available cache memory space in vehicles. This mechanism firstly introduces the concepts of load space and available load spcae, defined as:

Definition 1: The load capacity $\mathcal{L}_c$ of one node is the product of its storage capacity $\mathcal{S}_c$ and its transmission capacity r_c, which can be briefly defined as follows:

$$\mathcal{L}_c = \mathcal{S}_c \cdot r_c \tag{15}$$

Definition 2: The available load capacity $\Delta\mathcal{L}_c$ of one node is the gap between the total capacity and current used capacity. Thus, it can be briefly represented as follows:

$$\Delta\mathcal{L}_c = \left(\bar{S}_c - {S_c}'\right) \cdot \left(\bar{r}_c - {r_c}'\right) \tag{16}$$

where $\bar{\mathcal{S}}_c$ and $\bar{r}_c$ are the total storage size and upload bandwidth; and ${\mathcal{S}_c}'$ and ${r_c}'$ are the used ones.

As mentioned before, when no enough upload bandwidth is available even using both transmission paths, the cooperative caching is triggered. IC-Cloud first determines the proper cooperative caching nodes by calculating the available load space for the nearby nodes of the requesting node. Then, IC-Cloud assigns each neighbor node a probability to execute in-network caching operations that increases with the available load space. This metric indicates the priority level of one node conducting cooperative caching.

On the other hand, IC-Cloud should calculate the required number of cooperative nodes. To save the available caching resources, we consider a limiting condition to estimate the least required number of caching nodes.

Condition 2: To determine the required number of caching nodes, we propose that all requests for a video segment should be answered within a tolerable waiting time τ_l. The value of τ_l is dependent on the playback buffer size and the video playback rate, which avoids the video playback freeze and ensures the video be played smoothly. Considering there are different average service rate for two kinds of communications paths μ_I and μ_V, the total number of providers should satisfy the following condition:

$$(\mu_I + \mathcal{N}_p \cdot \mu_V) \cdot \tau_l \geq \mathcal{R}_i \tag{17}$$

where $\mathcal{R}_i$ is the number of requests for a certain video segment M_i, $\mathcal{N}_p$ is the total required number of providers using V2V/R transmission faces.

Let $\alpha = \frac{\mu_V}{\mu_I}$, we can obtain a metric for the least required number of vehicle providers as follows:

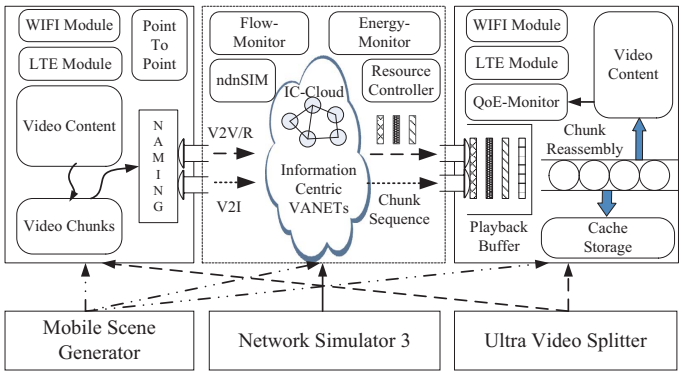


Fig. 6. A brief illustration of simulation platform.

$$\mathcal{N}_p \geq \frac{\mathcal{R}_i}{\mu_V \tau_l} - \frac{1}{\alpha} \tag{18}$$

Based on this, the required number of cooperative caching nodes $\mathcal{N}_{cc}$ for a certain multimedia can be calculated as the following:

$$\mathcal{N}_{cc} \geq \mathcal{N}_p - \bar{\mathcal{N}}_p \tag{19}$$

where $\bar{\mathcal{N}}_p$ is the number of current efficient providers.

Up to now, IC-Cloud gets to know the number of cooperative caching nodes, and then determines the proper cooperative nodes according to sorting the available load space. The nodes with higher available load space will be preferred in our design. As for what contents to be cached, IC-Cloud further gets to deduce which segments having great probability as the following segments to be requested according to a statistical regularity [44]. Algorithm 3 summarizes the cooperative in-network caching process in GrIMS. Specifically, the cooperative caching process includes the following features: (*i*) the popular contents trigger this cooperative in-network caching; (*ii*) the least number of caching nodes are selected to achieve the required service capacity; (*iii*) the selected nodes actively retrieve and cache the contents that have a higher chance to be requested in the next future.

VI. Performance Evaluation

A. Simulation Settings

The network platform is built based on the Network Simulator (NS-3) [47]. We merge and patch many powerful tools to build our experimental platform. To evaluate user's QoE levels, we adopt the special evaluation tool named QoE monitor [48]. The built-in energy model in NS-3 is used to monitor the energy consumption. The ndnSim [49] is adopted to implement the information-centric delivery function. As for the forwarding strategies, we do not directly adopt the default ones in ndnSim, such as Best Route Strategy, Broadcast Strategy and Client Control Strategy. Instead, we modify the Client Control Strategy by implementing a global resource controller, which is to undertake the IC-Cloud related functions to control and select the forwarding faces. Besides, we apply many other necessary module libraries, which are listed in Fig. 6 due to the space limitation. Many patches are also developed in the C++ programming language to make them work together well. Fig. 6 shows a brief illustration of our simulation platform.

Fig. 7. (a) The realistic street scenario (b) simulated scenario (400 nodes)

TABLE III
Video Samples Characteristics

Video	Data Rate	Resolution	Frame Rate	Codecs	Format
ICE 4CIF	1536 kbps	720 × 576	60 *fps*	H.264	AVI
ICE CIF	768 kbps	320 × 240	30 *fps*	H.264	AVI
ICE QCIF	256 kbps	176 × 144	15 *fps*	H.264	AVI

To simulate a realistic street scenario, we select the Financial Street Shopping Center (FSSC) in a big city (as shown in Fig. 7(a)), which has an approximate area of 1.8×1.8 km^2, and includes four major streets intersecting other four major ones. The VanetMobiSim [45] and BonnMotion [46], mobility scenario generation and analysis tools, are used to simulate the mobility environment, where the Manhattan Grid mobility model is adopted. In the simulation, we consider an urban VANETs scenario where 100-1600 vehicles drive along the street, the speed varies from 10km/h to 60km/h. 40 RSUs are deployed every 480m along the street, and 4 LTE BSs are located in the four corners of the area. A media server connects with all RSUs by wired connections of 1 Gbps, while with all BSs by wired connections of 10 Gbps. Fig. 7(b) shows a snapshoot of our simulated mobility scenario.

As for the testing videos, the video samples are downloaded from the well-known video website [50]. In our experiments, the "ICE" video sample is adopted in three different resolutions: ICE QCIF (176×144), ICE CIF (320×240), ICE 4CIF (720×576). Since the original video sequences downloaded are in the uncompressed yuv4mpeg format, we encode them uniformly with the H.264 video codec and produce a set of corresponding videos in three different data rates (256 kbps, 768 kbps and 1536 kbps). Then, each video is further split into many unified video chunks with a constant length. We adopt *UltraVideoSplitter* tool [51] to split each video into several 1s-duration video segments. The last segment of each video may be shorter than 1 second. Each video chunk is uniformly named by the format of *video number_resolution_chunk number*, for instance *ice1_CIF_1*, *ice5_QCIF_3*. This naming format is in accordance with the one in ndnSim. More detailed characteristics about the videos are listed in Table III.

Initially, the video segments are randomly distributed in all vehicles, and random 10% of total vehicles start to request the video contents. The requests of multimedia contents follow an MMRP distribution [31]. Note that a video request is started as the request of the first chunk of the video. Once a chunk is received, the next chunk request is emitted until the reception of the last chunk of the video [32]. All received video chunks will be reassembled in the receiver to generate a whole video

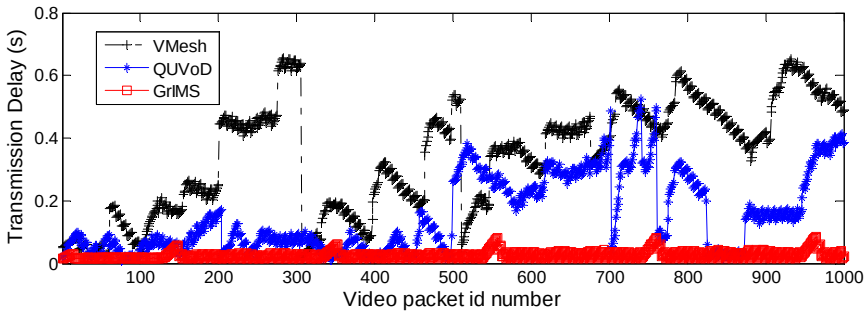


Fig. 8. Packet transmission delay during a video transmitting procedure

TABLE IV
MAIN PARAMETERS SETTING IN SIMULATION

Parameters		Values
General	Number of vehicles	100-1600
	Number of RSUs	40
	Number of LTE BSs	4
	Mobility model	Manhattan Grid model
	Peak Mobility Speed	10-60 km/h
	Wired bandwidth	1,10 $Gbps$
	Delay	10 ms
	Max Queue Len	500
WiFi	Operating Frequency	2.407 GHz
	PhyMode	DsssRate11Mbps
	Fragmentation Threshold	2200
	RtsCts Threshold	2200
	Wifi-Phy-Standard	802.11b
	Wifi-Mac-Type	AdhocWifiMac
	Transmission range	250 m
	PropagationDelayModel	ConstantSpeedPropagationDelay
	PropagationLossModel	FriisPropagationLossModel
	TxPower	16.0208 db
	EnergyDetectionThreshold	-71.9842
	Operating Voltage	3 V
	Current (Tx,Rx,Idle,Sleeping)	17.4, 19.7, 0.42 mA, 3 μA
LTE	Operating Frequency	2.5 GHz
	LteEnbPhy-TxPower	43 db
	LteUePhy-TxPower	40 db
	LteAmc-AmcModel	PiroEW2010
	LteRlcUm-MaxTxBufferSize	20971520 Bytes
	Transmission Mode	MIMO Tx Diversity
	SrsPeriodicity	320 ms
	Dl/UlEarfcn[1]	10, 18100
	Dl/UlBandwidth[2]	100
	Operating Voltage	3.5 V
	Current (Tx,Rx,Idle,Sleeping)	100, 105, 1 mA, 6 μA

[33]. Each vehicular node is equipped with a playback buffer, which is used to store the video data to be played. The buffer size is dependent on the selected player. In this experiment, we consider the buffer in each node contains 5 video segments at most. When the video data are played, they will be removed in a sequence. Each node also maintains a cache storage to cache the passing data, and LRU caching/replacement policy is adopted. These cached data make the node as the potential video provider in the following requests.

In terms of the network configuration, each vehicular node is equipped with two kinds of wireless interfaces, where the IEEE 802.11p WAVE is used to provide V2V communications, and LTE protocol for V2I communications. NS-3 [47] supports a node to be equipped with multiple interfaces. In our simulations, we use the built-in *WifiHelper* and *LteHelper* to configure the two protocols respectively. Some parameters adopted in our simulations are listed in the Table IV. Energy parameters are set according to the actual interfaces with the standard of IEEE 802.11p and LTE. We trace the behaviors and status in each device, calculate the energy consumption and record the received packets for the following analysis.

Two state of the art schemes are compared: VMesh [52] and QUVoD [12] in terms of the QoS-based metrics, the QoE-related metrics as well as the energy-related metrics. These metrics include average start-up delay, frequency of play disruption, average video quality (PSNR, SSIM, VQM), energy consumption and energy-to-qoe ratio (EQR). The experiment cases are repeated for 100 times, and the average results and 95% confidence interval are calculated.

B. Simulation Results

(1) QoS-based metrics:

We focus on the transmission delay, jitter and average throughput in terms of QoS-based metrics. Fig. 8 shows the packet transmission delay during a video transmitting procedure. In this figure, we can see VMesh and QUVoD have big transmission delay and jitter, while GrIMS has small ones. This is due to the multi-path selection and cooperative in-network caching mechanisms employed, which provide relative reliable transmitting paths and resource providers. What's more, we find that the delay increases as the packet id number increases in VMesh and QUVoD. That's because the number of video requesters increases with the time. In GrIMS, this change is not obvious. It is because GrIMS enables to adjust the available resources adaptively by the cooperative caching, which ensures a stable transmission quality, even though the video requesters increases.

We further analyze the transmission jitter and the average throughput in cases of different number of vehicles. Fig. 9 shows the jitter results in three solutions (VMesh, QUVoD, GrIMS). We observe VMesh has a relative large jitter, which increases from 12ms to 32ms with the increase of number of vehicles, ranging from 600 to 1600. QUVoD has a better performance than VMesh, where the jitter varies from 7ms to 22ms. In compared with above two, GrIMS achieves the least transmission jitter. What's more, its jitter maintains in a small range, affected a little by the number of vehicles. That's because the multipath selection ensures the reliable transmitting, and cooperative in-network caching further adds suitable resource providers. Fig. 10 shows the throughput conditions in three solutions. In this figure, the throughput decreases as the increase of number of vehicles. The main reason is that more video requesters will occupy more available bandwidth

[1]Uplink/Downlink E-UTRA absolte radio frequency channel number.

[2]Uplink/Downlink transmission bandwidth configuration in number of resource blocks.

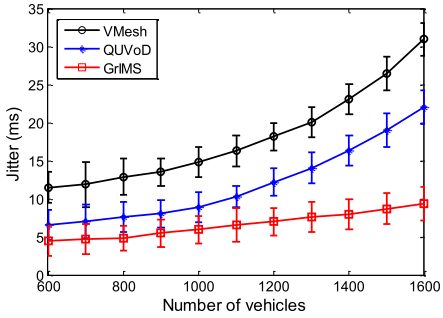


Fig. 9. Jitter vs. Number of vehicles

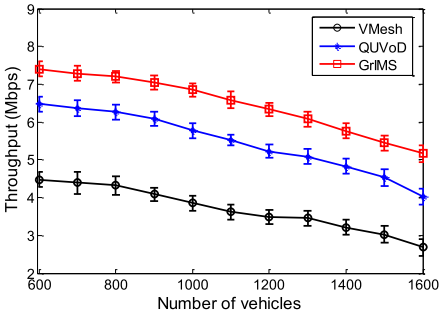


Fig. 10. Throughput vs. Number of vehicles

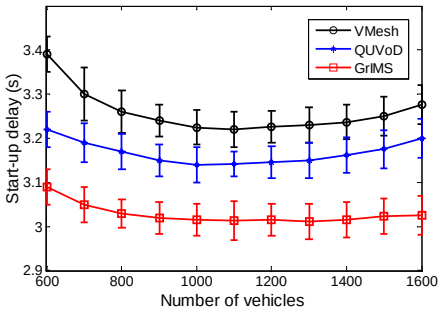


Fig. 11. Start-up delay vs. Number of vehicles

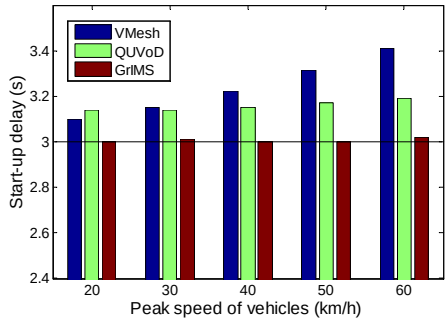


Fig. 12. Start-up delay vs. Peak speed of vehicles

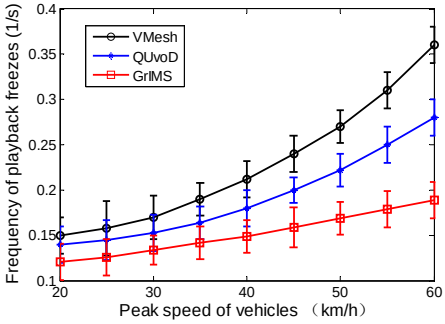


Fig. 13. Freezes vs. Peak speed of vehicles

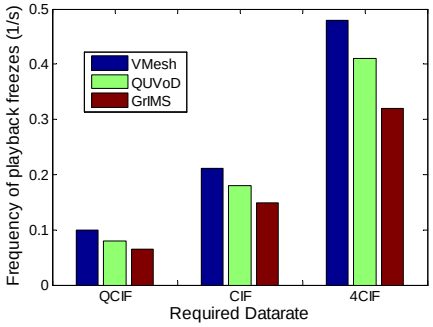


Fig. 14. Freezes vs. Required datarate

resources as the number of vehicles increases. GrIMS achieves a better throughput than the two other solutions.

Start-up delay is an important metric for multimedia delivery, which is defined as the waiting time between sending the Interest packets and starting to play the video. In this experiment, we set the video starts to play when the first 3s long video chunks are buffered. Due to the video resolution will affect the start-up delay greatly, we use the 4CIF format videos in this experiment. The peak mobility speed of vehicles is fixed at 40km/h. Fig. 11 shows that the start-up delay varies with the number of vehicles. It is interesting that the start-up delay in all three solutions first decreases with the increase of number of nodes, and then increase slightly. It is because that when the number of vehicles is small, increasing vehicles can help other peers to find a satisfied resource easier. However, when the number of vehicles reaches a certain level, these vehicles start to compete for the limited bandwidth. Noteworthy is that the solution GrIMS has a lower start-up delay than the two existing proposals in different scale cases. The reason mainly lies in two aspects: (i) the cooperative in-network caching improves the utilization of free nodes, producing more potential content providers; (ii) the multi-path transmission provide more reliable transmission to reduce delay of content retrieval.

Fig. 12 shows the movement influence on the start-up delay, where the mobility speed is ranging from 20km/h to 60km/h. The number of vehicular nodes is set to 1000. We can observe the start-up delays of GrIMS are always maintained at low levels. It is because that GrIMS adopts a reliable IC-Cloud to decide how to deliver the requested content, and the adaptive decision is available to improve the performance of delivery.

(2) QoE-related metrics:

Playback freeze is a metric for the playback continuity. We compute the playback freeze frequency by recording the time when the play buffer is empty during the playback process. In each vehicle peer, a special algorithm is used to compute to its freeze rate. Once a video is requested by a vehicle peer v, it starts to record the time T_1 in v when its play buffer is empty during the playback process. When the video is played over, it will record the total consumed time T, and then compute the playback freeze frequency by dividing the T_1 by the total consumed time T. It is noted that T_1 computing is triggered by an empty buffer and ended by a new buffered chunk.

Fig. 13 illustrates the frequency of playback freeze comparison results for different ranges of mobility speeds with a fixed 1000 vehicular nodes (in case of 4CIF format videos). As the figure shows, when nodes have low mobility speed (less than 30km/h), VMesh and QUVoD perform well. However, the freeze frequency of VMesh increases greatly with the speed, the playback freezes reach to 36% when the speed is up to 60km/h. On the other hand, the GrIMS's frequency of playback freeze is maintained at a low level, lower than that of VMesh by about 19-47% and lower than that of QUVoD by about 13-32%. In VMesh, the data transmitted via V2V path is not reliable in high mobility circumstances, so the performance of VMesh is the worst of all. In GrIMS, we use the information-centric delivery to make sure the user gets the video anywhere and anytime. Additionally, the enhanced mechanisms proposed improve the playback continuity.

Fig. 14 plots the comparison results when QUVoD, VMesh and GrIMS are used in turn with different QoE requirements. We consider three video formats with different levels of data rate: 256 kbps (QCIF), 768 kbps (CIF), 1536 kbps (4CIF), respectively. In this experiment, the mobility speed is fixed at the 40km/h and the system scale is set to 1000 vehicular nodes. As figure shows, the frequency of playback freeze in VMesh is much higher than any other solutions, especially in a high data rate, it freezes average 48% when the requested video is in

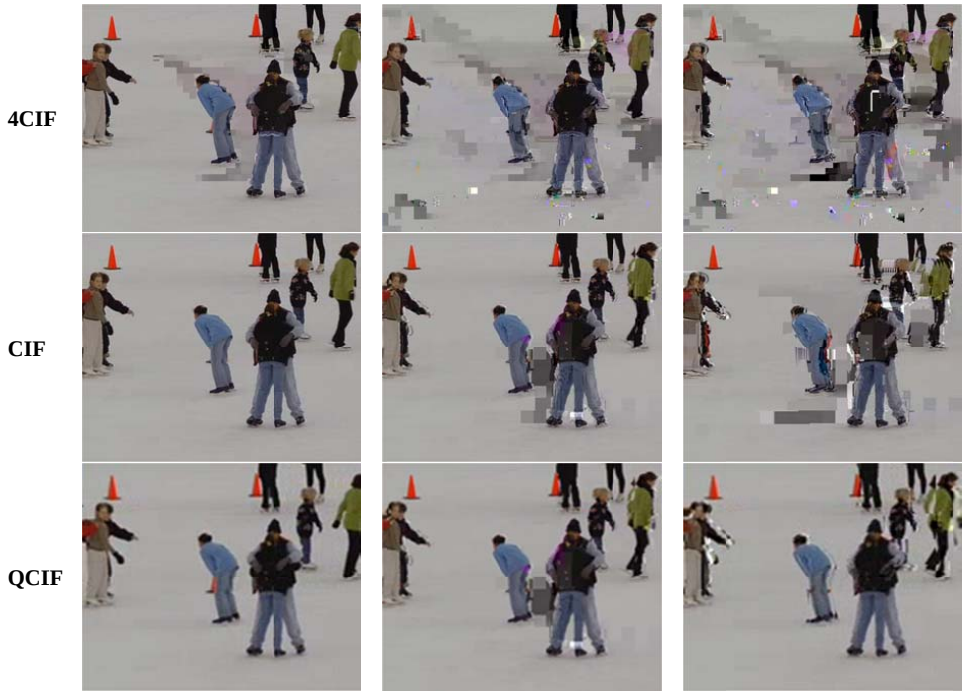


Fig. 15. Frames taken from received and reconstructed videos with three different formats by (a) GrIMS (b) QUVoD (c) VMesh

TABLE V
COMPARISON RESULT OF AVERAGE VIDEO QUALITY

Speed km/h	Methods	PSNR(dB)	SSIM
20 km/h	VMesh	26.45	0.8310
	QUVoD	34.62	0.9150
	GrIMS	46.48	0.9698
40 km/h	VMesh	19.58	0.8045
	QUVoD	27.00	0.8624
	GrIMS	42.93	0.9518
60 km/h	VMesh	10.46	0.7874
	QUVoD	20.63	0.8286
	GrIMS	38.49	0.9336

4CIF format. This is because the unreliable transmission path employed in VMesh suffers high packet loss rate when the data rate at high levels. QUVoD performs better than VMesh because the nodes can transfer the video content via the more stable 4G network path. The GrIMS performs best due to the efficient reliable delivery mechanisms, which enable to provide a high level of QoE.

In order to evaluate the Video Quality Measurement (VQM), we further analyze the received video by users in terms of PSNR and SSIM by using of the QoE monitor tool. Fig. 15 shows the snapshot comparison of received videos by three solutions. In this figure, the mobility speed was set to 60 km/h and the number of vehicular nodes is 1000. We observe that QUVoD and VMesh obtain poor video frames especial for the high resolution formats. GrIMS achieves relative high quality frames, and provide a good quality of experience for users (especial with the CIF and QCIF format videos). That is because GrIMS employs the cloud-based processing, QoE-guaranteed multi-path transmission and cooperative in-network caching to improve the transmission quality and enhance users' QoE levels. The figure clearly illustrates GrIMS outperforms the other solutions in terms of perceived quality.

Table V presents other comparison results of average video quality, expressed in terms of PSNR (dB), SSIM. The mobility speed varies from 20 *km/h* to 60 *km/h*. As the table illustrates, GrIMS outperforms VMesh and QUVoD, especially the speed is 20 *km/h*, the average PSNR of VMesh and QUVoD are 26.45 and 34.62, respectively, and PSNR of GrIMS is 46.48, improving by 34%-75%. However, when the speed reaches 60 *km/h*, PSNR of VMesh drops to 10.46, that of QUVoD is 20.63, and PSNR of GrIMS is as high as 38.49. This can be explained by the benefits of using the novel cloud-based computing mechanism and multi-path selection solution which help users get better suppliers and delivery paths. Besides, cooperative in-network caching can further ensure the quality of service by making using of extra resources. The same experiment results are presented in terms of SSIM. These results confirm that GrIMS outperforms VMesh and QUVoD in the QoE of received videos.

(3) Energy consumption metrics:

In our experiment, we calculate a summation of *energy consumption* of all devices (including both base stations and mobile devices) during the entire simulation process as the one of the entire network. The energy consumption for each device is calculated by the built-in energy module. The computational formula is as follows: $energy\ consumption = current \times voltage \times time$. Fig. 16, 17 and 18 show the results of energy consumption using different solutions.

Fig. 16 focuses on energy consumption in LTE communications. We observe that the energy consumption increases as the increase of the number of vehicles. GrIMS consumes less energy in LTE devices than QUVoD. The main reason is that the enhanced mechanisms employed in GrIMS can promote the utilization of V2V paths, and reduce the probability of expensive LTE communications. Fig. 17 shows the results

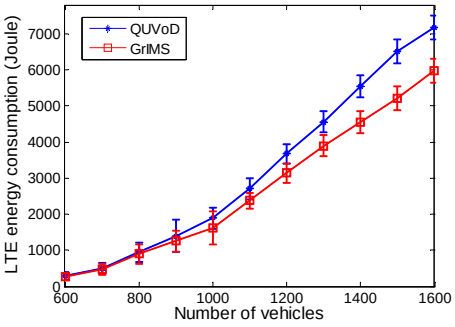


Fig. 16. Energy consumption in LTE devices

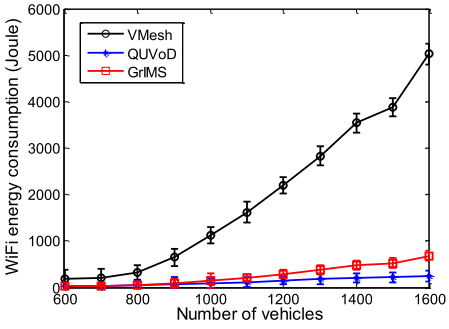


Fig. 17. Energy consumption in WiFi devices

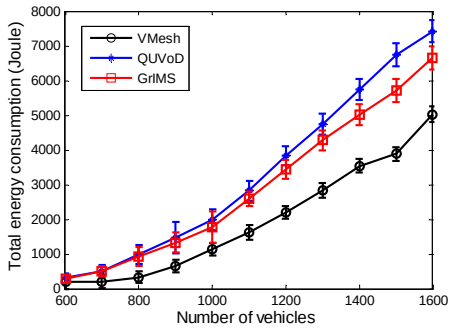


Fig. 18. Total energy consumption

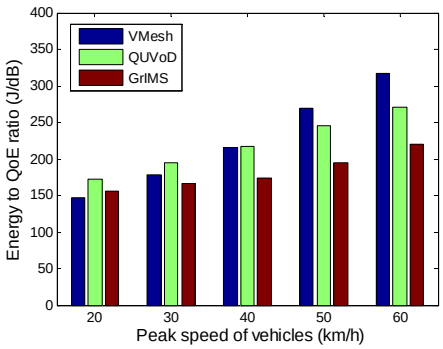


Fig. 19. Energy consumption to QoE ratio

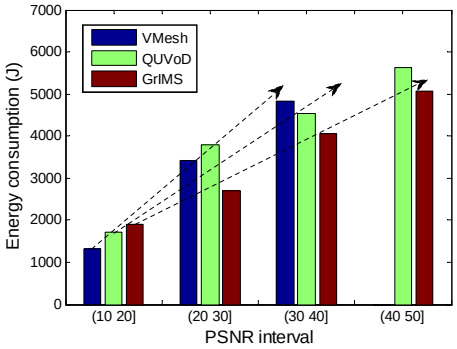


Fig. 20. Energy consumption vs. PSNR intervals

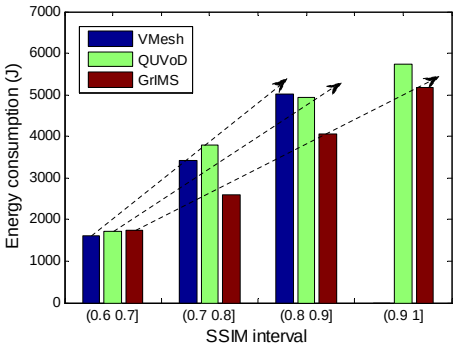


Fig. 21. Energy consumption vs. SSIM intervals

of energy consumption in WiFi communications. We find that GrIMS consumes a bit more than QUVoD, and VMesh costs the most energy in WiFi communications. In VMesh, all transmissions adopt the WiFi communications, and the utilization of WiFi devices determine the energy consumption amount. Fig. 18 compares the total energy consumption of the entire system in three different solutions. It shows that GrIMS cost less energy than QUVoD, while consumes a bit more energy than VMesh. Note that this experiment only studies the energy consumption varying with the number of vehicles, does not consider the quality of video delivery.

In order to further analyze energy efficiency in different solutions, we test to compare the energy consumption of different solutions with same QoE levels. However, the QoE level is very difficult to be pre-controlled in practical experiment due to the changeable environment. Thus, a novel alternative method is to use the Energy to QoE Ratio (EQR), $EQR = energy\ consumption/QoE$, which denotes the energy consumption relative to the QoE value for a complete multimedia delivery. As for the QoE value, we adopt $PSNR \times SSIM$ as a straightforward evaluation metric of QoE level. Fig. 19 shows the EQR varies with the mobile speeds in different solutions. The figure shows GrIMS has the least EQR among all three solutions. VMesh achieves a slightly superiority than QUVoD when the speed is smaller than 40 km/h, while this situation changes when the speed exceeds 40 km/h. It is easily understood that QUVoD achieves a better QoE than VMesh, and makes a sacrifice in term of energy consumption due to the additional maintain overhead. As the increase of speed, the superiority of QUVoD in the QoE increases, thus, it brings a small increase rate of EQR. In the case of 60 km/h, GrIMS makes an obvious improvement in saving energy. The results also tell EQR value increases as the mobile speed increases, since a high speed brings the unreliable data transfer.

Besides, we analyze the energy efficiency in different solutions on conditions of certain PSNR and SSIM intervals. We first divide the PSNR range into 4 intervals: (10, 20], (20, 30], (30, 40] and (40, 50]. Through adjusting different network parameters, we collected a great number of experimental data, which record the energy consumption values and their corresponding PSNRs. Then, we make a statistic analysis and calculate the mean of all energy consumption values within the same PSNR interval. It is approximately set as the average energy consumption within this PSNR interval. Fig. 20 shows the relation between energy consumption and PSNR. We find that GrIMS consumes the least energy among the three solutions in the cases of high PSNRs. Moreover, GrIMS has a smaller increasing rate of energy consumption than either of the other two solutions. Note that no data corresponds to VMesh-(40, 50] due to poor delivery quality in VMesh. Similar results are obtained in cases of SSIM intervals, which are shown in Fig. 21. The main reason is that multi-path transmission decision can guarantee the quality of video. On the other hand, in-network caching enables to increase potential content providers and reduce the hops of content delivery, hence, achieve energy savings.

All experiments results and above anysis confirm that GrIMS achieves a great energy saving while guaranteeing the quality of video transmission. It is noted that although some more energy is consumed in GrIMS, the Energy to QoE ratio can reduce, because it improves the video quality of experience.

VII. Conclusions

Multimedia delivery over VANETs is a promising area in future mobile networks. However, it remains an open issue

how to provide high quality video delivery with less energy consumption in VANET environment. In this paper, a novel Green Information-centric Multimedia Streaming framework for vehicular networks (GrIMS) was proposed. We adopted a queuing model with differentiated service rates to characterize the multimedia retrieval in heterogeneous wireless VANETs and formulated an optimization framework to understand the impact of GrIMS delivery solutions on both QoE and energy consumption. Furthermore, we proposed three heuristic mechanisms to achieve both high QoE levels and energy saving in GrIMS's cloud-based processing, multi-path selection and in-network caching context. Extensive tests have shown GrIMS provides considerable benefits in terms of average transmission delay, jitter, throughput, start-up delay, playback continuity, estimated user perceived quality and energy saving for multimedia delivery in highly dynamic VANETs.

REFERENCES

[1] L. Sarakis, T. Orphanoudakis, H. Leligou and S. Voliotis, "Providing Entertainment Applications in VANET Environments," *IEEE Wireless Communications*, vol. 23, no. 1, pp. 30-37, 2016.

[2] C. Xu, S. Jia, M. Wang, L. Zhong and G.-M. Muntean, "Performance-Aware Mobile Community-Based VoD Streaming Over Vehicular Ad Hoc Networks," *IEEE Trans. on Vehicular Technology*, vol. 64, no. 3, pp. 1201-1217, March 2015.

[3] L. Zhou, Z. Yang, Y. Wen and J. J. P. C. Rodrigues, "Distributed Wireless Video Scheduling with Delayed Control Information," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 24, no. 5, pp. 889-901, May 2014.

[4] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher and B. Ohlman, "A Survey of Information-centric Networking," *IEEE Commun. Magazine*, vol. 50, no. 7, pp. 26-36, July 2012.

[5] A. Detti, B. Ricci and N. Blefari-Melazzi, "Mobile peer-to-peer video streaming over information-centric networks," Computer Network, vol. 81, pp. 272-288, Apr. 2015.

[6] S. Lederer, C. Mueller, C. Timmerer and H. Hellwagner, "Adaptive multimedia streaming in information-centric networks," IEEE Network, vol. 28, no. 6, pp. 91-96, Nov. 2014.

[7] G. Tyson, N. Sastry, R. Cuevas, I. Rimac and A. Mauthe, "A Survey of Mobility in Information-Centric Networks," *Communications of the ACM*, vol. 56, no. 12, pp. 90-98, Dec. 2013.

[8] F. Bai, B. Krishnamachari, "Exploiting the Wisdom of the Crowd: Localized, Distributed Information-Centric VANETs," IEEE Communications Magazine, vol. 48, no. 5, pp. 138-146, May 2010.

[9] G. Piro, L. A. Grieco, G. Boggia and P. Chatzimisios, "Information-centric Networking and Multimedia Services: Present and Future Challenges," *Trans. on Emerging Tel. Tech.*, pp. 1-15, Sep. 2013.

[10] M. Amadeo, C. Campolo and A. Molinaro, "Content-Centric Networking: Is That a Solution for Upcoming Vehicular Networks," *Proc. of ACM internationl workshop on VANET*, 2012.

[11] G. Arnould, D. Khadraoui and Z. Habbas, "A Self-Organizing Content Centric Network Model for Hybrid Vehicular Ad-Hoc Networks," *Proc. of ACM DIVANet*, 2011.

[12] P. TalebiFard and V. Leung, "A Content Centric Approach to Dissemination of Information in Vehicular Networks," *Proc. of ACM DIVANet*, 2012.

[13] L. Wang, R. Wakikawa, R. Kuntz, *et al.*, "Data Naming in Vehicle-to-Vehicle Communications," *Proc. of IEEE INFOCOM workshop*, 2012.

[14] A. Detti, M. Pomposini, N. BlefariMelazzi and S. Salsano, "Offloading Cellular Networks with Information-Centric Networking: The Case of Video Streaming," *Proc. of IEEE WoWMoM*, 2012.

[15] Z. Li and G. Simon, "Time-Shifted TV in Content Centric Networks: the Case for Cooperative In-Network Caching," *Proc. of IEEE ICC*, 2011.

[16] L. Zhou, R. Hu, Y. Qian, *et al.*, "Energy-Spectrum Efficiency Tradeoff for Video Streaming over Mobile Ad Hoc Networks," *IEEE Journal on Selected Areas in Commun.*, vol. 31, no. 5, pp. 981-991, May 2013.

[17] M. S. Obaidat, A. Anpalagan, I. Woungang, "Handbook of Green Information and Communication Systems," *Academic Press*, Nov. 2012.

[18] C. Xu, S. Jia, L. Zhong, and G.-M. Muntean, "Socially Aware Mobile Peer-to-Peer Communications for Community Multimedia Streaming Services," *IEEE Commun. Magazine*, vol. 53, no.10, pp. 150-156, 2015.

[19] F. Malandrino, C. Casetti, C.-F. Chiasserini and M. Fiore, "Optimal Content Downloading in Vehicular Networks," *IEEE Trans. Mobile Computing*, vol. 12, no. 7, pp. 1377-1391, Jul. 2013.

[20] O. Trullols-Cruces, M. Fiore and J. M. Barcelo-Ordinas, "Cooperative Download in Vehicular Environments," *IEEE Trans. Mobile Computing*, vol. 11, no. 4, pp. 663-678, Apr. 2012.

[21] L. Zhou, Y. Zhang, K. Song, W. Jing and A.-V. Vasilakos, "Distributed Media Services in P2P-Based Vehicular Networks," *IEEE Trans. Veh. Technol.*, vol. 60, no. 2, pp. 692-703, Feb. 2011.

[22] N. N. Qadri, M. Fleury, B. R. Rofoee, M. Altaf and M. Ghanbari, "Robust P2P Multimedia Exchange within a VANET," *Wireless Personal Commun.*, vol.63, no.3, pp. 561-577, Apr. 2012.

[23] C. Xu, F. Zhao, J. Guan, *et al.*, "QoE-driven User-centric VoD Services in Urban Multihomed P2P-based Vehicular Networks," *IEEE Trans. Veh. Technol.*, vol. 62, no. 5, pp. 2273-2289, Jun. 2013.

[24] D. Feng, C. Jiang, G. Lim, *et al.*, "A Survey of Energy-efficient Wireless Communications," *IEEE Commun. Surveys and Tutorials*, vol. 15, no. 1, pp. 167-178, 2013.

[25] M. Arumaithurai, K. Ramakrishnan, *et al.*, "Information Centric Networking: The Case for An Energy Efficient Future Internet Architecture," *Book chapter in Green Communications: Principles, Concepts and Practice*, Wiley, 2015.

[26] K. Pentikousis, "In search of energy-efficient mobile networking ," *IEEE Commun. Magazine*, vol.48, no. 1, pp. 95-103, 2010.

[27] G. Piro, M. Amadeo, G. Boggia, *et al.*, "Gazing into the Crystal Ball: When the Future Internet Meets the Mobile Clouds," *IEEE Trans. on Cloud Computing*, vol. PP, no. 99, pp. 1-1, 2016.

[28] U. Lee, I. Rimac and V. Hilt, "Greening the Internet with Content-centric Networking," *Proc. of ACM Energy-Efficient Computing and Networking*, 2010.

[29] M. Butt, O. Delgado and M. Coates, "An Energy-efficiency Assessment of Content Centric Networking," *Proc. of IEEE CCECE*, 2012.

[30] N. Choi, K. Guan, *et al.*, "In-Network Caching Effect on Optimal Energy Consumption in Content-Centric Networking," *Proc. of IEEE ICC*, 2012.

[31] T. E. Stern and A. I. Elwalid, "Analysis of Separable Markov-modulated Rate Models for Information-handling Systems," Advances in Applied Probability, vol. 23, no. 1, pp. 105C139, 1991.

[32] L. Zhang, D. Estrin, V. Jacobson, *et al.*, "Named Data Networking (NDN) project," *Technical Report*, NDN-0001, 2010.

[33] 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. 12, no. 11, pp. 2193-2205, Sep. 2013.

[34] G. Garofiglio, M. Gallo, L. Muscariello *et al.*, "Modeling Data Transfer in Content-centric Networking (extended version)," Orange Labs/Alcatel-Lucent Technology Report, 2011.

[35] E. Lee, E. Lee, M. Gerla and S. Oh, "Vehicular Cloud Networking: Architecture and Design Principles," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 148-155, 2014.

[36] F. Hermans, E. Ngai and P. Gunningberg, "Global Source Mobility in the Centent-Centric Networking Architecture," *ACM NoM*, 2012.

[37] Amazon EC2, *http://aws.amazon.com/ec2/*.

[38] T. C. Mills, "Time Series Techniques for Economists," *Cambridge University Press*, 1991.

[39] Y.-T. Chan and J.-C. Wood, "A New Order Determination Technique for ARMA Processes," *IEEE Trans. Acoustics, Speech, Signal Processing*, vol. 32, no. 3, pp. 517-521, June 1984.

[40] W. Quan, C. Xu, J. Guan, *et al.*, "Scalable Name Lookup with Adaptive Prefix Bloom Filter for Named Data Networking," *IEEE Communications Letters*, vol. 18, no. 1, pp. 102-105, Jan. 2014.

[41] W. Quan, C. Xu, A. V. Vasilakos, *et al.*, "TB2F: Tree-Bitmap and Bloom-Filter for a Scalable and Efficient Name Lookup in Content-Centric Networking," *IFIP Networking*, Jun. 2014.

[42] C. Xu, Z. Li, J. Li, H. Zhang and G.-M. Muntean, "Cross-layer Fairness-driven Concurrent Multipath Video Delivery over Heterogenous Wireless Networks," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 25, no. 7, pp. 1175-1189, July 2015.

[43] C. Xu, Z. Li, L. Zhong, H. Zhang and G.-M. Muntean, "CMT-NC: Improving the Concurrent Multipath Transfer Performance of SCTP using Network Coding in Wireless Networks," *IEEE Trans. on Vehicular Technology*, vol. 65, no. 3, pp. 1735-1751, March 2016.

[44] V. Krishnamoorth, N. Carlsson and D. Eager, "Quality-adaptive Prefetching for Interactive Branched Video using HTTP-based Adaptive Streaming," *ACM Multimedia*, Nov. 2014.

[45] VanetMobiSim, *http://vanet.eurecom.fr*.

[46] Bonnmotion, *http://sys.cs.uos.de/bonnmotion/index.shtml*.

[47] NS-3, *http://www.nsnam.org*.

[48] QoE monitor, *http://sourceforge.net/projects/ns3qoemonitor.*
[49] ndnSim in NS-3, *http://ndnsim.net/intro.html.*
[50] Xiph.org Video Test Media, *http://media.xiph.org/video/derf.*
[51] Aone-soft, *http://www.aone-soft.com/splitter.htm (2014).*
[52] W.-P. K Yiu, X. Jin and S.-H. G Chan, "VMesh: Distributed Segment Storage for Peer-to-Peer Interactive Video Streaming," *IEEE Journal on Selected Areas in Commun.*, vol. 25, no. 9, pp. 1717-1731, Dec. 2007.

APPENDIX A

According to the equation set (2), let Eq. (2.2) multiply by μ_I, and minus the Eq. (2.3) multiplied by μ_V, we can have:

$$\mu_I\left(\lambda+\mu_I\right)p_V-\mu_V\left(\lambda+\mu_V\right)p_I=\left(\left(1-\varphi\right)\mu_I-\varphi\,\mu_V\right)\lambda p_0 \tag{20}$$

Let Eq. (2.1) multiply by $(\lambda+\mu_V)$, then

$$\left(\lambda+\mu_V\right)\mu_I\,p_V+\left(\lambda+\mu_V\right)\mu_V\,p_I=\lambda\left(\lambda+\mu_V\right)p_0 \tag{21}$$

Combining the Eq. (20) with Eq. (21), then

$$\left(2\lambda+\mu\right)\mu_I\,p_V=\lambda\left(\lambda+\mu-\mu\varphi\right)p_0 \tag{22}$$

Clearly,

$$\begin{aligned} p_V &= \tfrac{\lambda\mu(\rho+1-\varphi)}{(2\lambda+\mu)\,\mu_V}\,p_0 \\ &= \tfrac{\rho}{1+2\,\rho}\cdot\tfrac{1+\alpha}{\alpha}\left(\rho+1-\varphi\right)p_0 \end{aligned} \tag{23}$$

In a similar way, we can have the relation:

$$\left(2\lambda+\mu\right)\mu_V\,p_I=\lambda\left(\lambda+\varphi\mu\right)p_0 \tag{24}$$

Then, we get

$$\begin{aligned} p_I &= \tfrac{\lambda(\lambda+\varphi\mu)}{(2\lambda+\mu)\,\mu_V}\,p_0 \\ &= \tfrac{\rho}{1+2\,\rho}\left(1+\alpha\right)\left(\rho+\varphi\right)p_0 \end{aligned} \tag{25}$$

Adding the Eq. (2.2) with Eq. (2.3), and combine with the Eq. (2.1), we have

$$\mu\,p_2=\lambda\,p_V+\lambda\,p_I \tag{26}$$

Then, we further get

$$\begin{aligned} p_2 &= \rho_1\,p_V+\rho_1\,p_I \\ &= \tfrac{\rho^2}{1+2\,\rho}\tfrac{1+\alpha}{\alpha}\left[1+\left(1+\alpha\right)\rho-(1-\alpha)\varphi\right]p_0 \end{aligned} \tag{27}$$

Combining Eq. (2.4) with Eq. (27),

$$\lambda\,p_2=\mu\,p_3 \tag{28}$$

Generally, we have

$$p_n=\frac{\rho^n}{1+2\,\rho}\frac{1+\alpha}{\alpha}\left[1+\left(1+\alpha\right)\rho-(1-\alpha)\varphi\right]p_0 \tag{29}$$

Then, we summarize the equation set of Eq. (3).

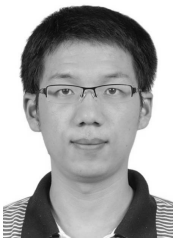
APPENDIX B

The derivation process is as follows:

$$\begin{aligned} &m_s=p_V+p_I+\sum_{n=2}^{\infty}n\,p_n \\ &=\tfrac{\rho}{1+2\,\rho}\tfrac{1+\alpha}{\alpha}\left(\rho+1-\varphi\right)p_0+\tfrac{\rho}{1+2\,\rho}\left(1+\alpha\right)\left(\rho+\varphi\right)p_0+ \\ &\tfrac{\rho(1+\alpha)}{\alpha(1+2\,\rho)}\left[1+\left(1+\alpha\right)\rho-(1-\alpha)\varphi\right]p_0\sum_{n=2}^{\infty}n\,\rho^{n-1} \\ &=\tfrac{p_0\,\rho_1(1+\alpha)}{\alpha(1+2\,\rho_1)}\left\{\rho_1+1-\varphi+\alpha\left(\rho_1+\varphi\right)+\right. \\ &\left.\left[1+\left(1+\alpha\right)\rho_1-\left(1-\alpha\right)\varphi\right]\tfrac{2\,\rho_1-\rho_1^2}{(1-\rho_1)^2}\right\} \\ &=\tfrac{\rho(1+\alpha)[1+(1+\alpha)\,\rho-(1-\alpha)\varphi]}{\alpha(1+2\,\rho)\,(1-\rho)^2}\cdot\tfrac{\alpha(1+2\,\rho)(1-\rho)}{\alpha(1+2\,\rho)+\rho[1+(1+\alpha^2)\,\rho-(1-\alpha^2)\varphi]} \\ &=\tfrac{\rho(1+\alpha)}{1-\rho}\tfrac{1+(1+\alpha)\,\rho-(1-\alpha)\varphi}{\alpha(1+2\,\rho)+\rho[1+(1+\alpha^2)\,\rho-(1-\alpha^2)\varphi]} \end{aligned} \tag{30}$$

Then, we conclude the Eq. (6).

Changqiao Xu received the Ph.D. degree from the Institute of Software, Chinese Academy of Sciences (ISCAS) in January 2009. He was an Assistant Research Fellow in ISCAS from 2002 to 2007. During 2007 - 2009, he worked as a Researcher with the Software Research Institute at Athlone Institute of Technology, Athlone, Ireland. He joined Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in December 2009. Currently, he is an Associate Professor with the State Key Laboratory of Networking and Switching Technology, and Director of the Next Generation Internet Technology Research Center at BUPT. He has published over 100 technical papers in prestigious international journals and conferences. His research interests include wireless networking, multimedia communications, and next generation Internet technology. He served as Co-Chair of IEEE MMTC Interest Group "Green Multimedia Communications" and Board Member of IEEE MMTC Services and Publicity. He is Senior member of IEEE.



Wei Quan received his Ph.D degree from Beijing University of Posts and Telecommunications (BUPT) in 2014, worked as a Postdoctor during 2014-2016 and currently works as a Lecturer at Beijing Jiaotong University (BJTU), China. His research interests include key technologies for future Internet, 5G network architecture, vehicular networks and Internet of energy. He has published more than 20 papers in prestigious international journals and conferences, including IEEE Wireless Communications Magazine, IEEE Network Magazine, IEEE Communications Letters, IFIP Networking, IEEE WCNC. He also serves as technical reviewers for some important international journals and conferences.

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 (BJTU), and in July 1994, he became a professor there. He currently directs a National Engineering Lab on Next Generation Internet in China. His research has resulted in many research papers, books, patents, systems and equipment in the areas of communications, computer networks. He is the author of eight books written in Chinese and the holder of more than 70 patents. He is Senior member of IEEE.

Luigi Alfredo Grieco received the Dr. Eng. degree (with honors) in electronic engineering from the Politecnico di Bari, Bari, Italy, in October 1999 and the Ph.D. degree in information engineering from the university di Lecce, Lecce, Italy, in December 2003. He is an Associate Professor of telecommunications at the Politecnico di Bari, Bari, Italy. Formerly, he was a Visiting Researcher with INRIA (Sophia Antipolis, France) in 2009 and with LAAS-CNRS (Toulouse, France) in 2013, working on Internet measurements and M2M systems, respectively. He has authored more than 100 scientific papers published in international journals and conference proceedings that gained more than 1000 citations. His main research interests include multimedia communications, quality of service in wireless networks, Internet of things (IoT), and future Internet. He serves as an Editor of the IEEE Trans. on Vehicular Technology and as the Executive Editor of the Transactions on Emerging Telecommunications Technologies (Wiley). Within the Internet Engineering Task Force and Internet Research Task Force, he is actively contributing to the definition of new standard protocols for industrial IoT applications and new standard architectures for tomorrow Information Centric Networking (ICN)-IoT systems.