Engineering a Distributed Infrastructure for Large-Scale Cost-Effective Content Dissemination over Urban Vehicular Networks

Tom H. Luan *Student Member, IEEE*, Lin X. Cai *Member, IEEE*, Jiming Chen *Senior Member, IEEE*, Xuemin (Sherman) Shen *Fellow, IEEE*, and Fan Bai *Member, IEEE*

***Abstract*—This work proposes a practical and cost-effective approach to construct a fully distributed roadside communication infrastructure to facilitate the localized content disseminations to vehicles in the urban area. The proposed infrastructure is composed of distributed light-weight low-cost devices called roadside buffers (RSBs), where each RSB has the limited buffer storage and is able to wirelessly transmit the cached contents to fast-moving vehicles. To enable the distributed RSBs working towards the global optimal performance (*e.g.*, minimal average file download delays), we propose a fully distributed algorithm to optimally determine the content replication strategy at RSBs. Specifically, we first develop a generic analytical model to evaluate the download delay of files, given the distribution of content files at RSBs. We then formulate the RSB content replication process as an optimization problem, and devise a fully distributed content replication scheme accordingly to enable vehicles to intelligently recommend the desirable content files to RSBs. The proposed infrastructure is designed to optimize the global network utility which accounts for the integrated download experience of users and the download demands of files. Using extensive simulations, we validate the effectiveness of the proposed infrastructure and show that the proposed distributed protocol can approach to the optimal performance and significantly outperform the traditional heuristics.**

***Index Terms*—Vehicular Network; Content Distribution; Random Walk.**

I. Introduction

In the last couple of years, vehicular networking and communications have been identified as a key enabling technology to make our daily life on-the-wheel safer, more efficient and comfort with ubiquitous broadband services [1]. While being actively pursued for years, the real-world large-scale deployment of vehicular communications, however, is still not practical and fraught with many fundamental challenges. This attributes to the lack of an efficient accessing approach on providing ubiquitous, high-rate yet low-cost connections to vehicles. Using traditional 3G/4G cellular networks, not only the aggregate bandwidth per user is very limited as a large number of users need to share wireless resource concurrently, but also the usage cost per user is high. An alternative approach is by exploring city-wide WiFi hotspots for high-rate services at the low price. However, sparsely distributed in the city [2], [3] with limited coverage individually, WiFi hotspots can hardly provide ubiquitous connectivity to vehicles. Moreover, originally designed for static indoor applications, WiFi hotspots are not optimized for highly mobile vehicular communications [4]. Another plausible solution is by exploring inter-vehicle communications. While collaborative inter-vehicular communications can boost the system capacity, purely relying on the vehicle-to-vehicle (V2V) communications is insufficient to provide the reliable and high-rate data services to users due to harsh channel conditions and unreliable inter-vehicle connections. As reported in [5], the throughput of inter-vehicular communications is observed to be at most one fifth of the throughput of vehicle-to-infrastructure communications. In a nutshell, in order to bring vehicular communications and networking from lab concept to commercial reality, a novel, practical and scalable solution which offsets the weaknesses of traditional accessing approaches is desirable.

As an effort towards this goal, in this work, we propose a practical approach on building a low-cost city-wide infrastructure to enable content distributions to vehicular mobile users. We argue that a large-scale communication infrastructure which is dedicated to vehicular communications with reserved resources (*e.g.*, storage and communication capacity) is essential to provide reliable and QoS guaranteed services, so as to make vehicular communications an alternative of ubiquitous broadband access. Note that to build a large-scale infrastructure is typically a daunting task, if not impossible, due to the high deployment and maintenance cost. We question on how to construct a *practical* and *scalable* content distribution infrastructure in the city, which not only bypasses physical installations and investment obstacles, but also incurs the minimum monetary wireless bandwidth expense for individual users. To attain this goal, we propose a low-cost, fully distributed and self-maintained infrastructure. In specific, the proposed infrastructure is composed of a multitude of wireless buffer devices deployed on the roadside, namely roadside buffers (RSBs). Each RSB is equipped with a wireless transceiver operating on the dedicated short-range commu-

Parts of the results in this paper have been presented in the IEEE International Conference on Sensing, Communication, and Networking (SECON), Salt Lake City, Utah, USA, June, 2011.

Tom H. Luan, Lin X. Cai, Xuemin (Sherman) Shen are with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: hluan@bbcr.uwaterloo.ca; lcai@bbcr.uwaterloo.ca; xshen@bbcr.uwaterloo.ca).

Jiming Chen is with the State Key Laboratory of Industrial Control Technology, Department of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China (e-mail: jmchen@iipc.zju.edu.cn).

Fan Bai is with General Motors Research Center, Warren, MI, 48090 USA (e-mail: fan.bai@gm.com).

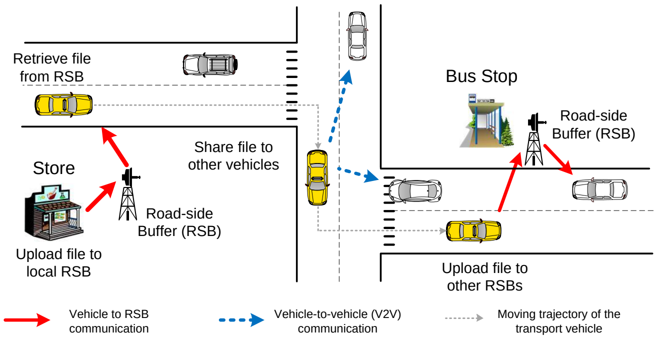


Fig. 1. Content distribution through the proposed infrastructure

nication (DSRC) radio, and can communicate with nearby vehicles using the vehicle-to-infrastructure communications. The RSB can selectively retrieves content files from the vehicle drive-through its coverage and disseminate the cached files to vehicles upon their requests. Fig. 1 shows a motivating example in which a grocery store intends to distribute its recent flyers to customers in the city. To do so, the flyers are first uploaded to one or multiple RSBs near the store. The RSBs are then responsible for distributing the content files (flyers) on the fly to vehicles driving through the area and let the vehicles spread the flyers to other RSBs and vehicles across the city.

Unlike the conventional centralized system (*e.g.*, cellular base stations), the proposed infrastructure (*i.e.*, RSBs) is distributedly deployed, owned and managed by separate entities. For example, a grocery store or shopping mall may deploy the RSB in its parking lot to periodically broadcast the flyers as in Fig. 1. A movie theater may install the RSB to distribute the latest movie tailors to the public. The distributed RSBs deployed by separate entities collectively form the infrastructure network. In other words, the formation of the proposed infrastructure relies on the contributions of separate individuals in the city with the shared investment and maintenance work of the devices. The distributed deployments of RSBs have the following features:

- ▷ Cheap and easy to install: the RSBs are cheap and lightweight devices composed of a wireless transceiver and small buffer. They can be configured and managed wirelessly, requiring no complex and expensive cabling work. As RSBs are deployed to distribute the local contents generated by their owners, they are not necessarily connected to the Internet. As such, once deployed, RSBs incur no bandwidth cost to their owners.
- ▷ Easy to manage: the content distribution and buffer management of RSBs are purely self-organized which autonomously adapt to the time-varying network conditions (*e.g.*, the density of vehicle traffic, buffer availability) [6] and are tailored to meet the file download demands. Therefore, except to using wireless connections, the owners are not required to get involved in any further operation.
- ▷ Profitable: The RSBs can bring commercial benefits to their owners by distributing the advertisements or other information to the public.

To summarize, the RSBs distributedly deployed in the city can provide dedicated storage and communication capacity to enable content distributions to vehicles. Moreover, by relying on the fast vehicles to transport contents among RSBs, and making RSBs selectively retrieve contents from vehicles to cache and redistribute, the entire infrastructure is designed to achieve a global optimal goal in a fully distributed manner.

The remainder of the paper is organized as follows: Section II describes related works, and position the original contributions of our work. Section III presents the system model and formulates the design of the network as an optimization problem. Section IV discusses on the solution of the problem and presents the network protocols. The performance of the proposed infrastructure is verified in Section V, and Section VI closes the paper with conclusions.

II. Related Works

In this section, we survey the related works and highlight our contributions in the light of existing literature.

The vehicular content distribution networks can be broadly categorized in two groups as: V2V based systems in which the content distribution mainly relies on the collaborations among vehicles using the V2V communications only [7], [8], [9], and the vehicle-to-infrastructure based systems which exploit the opportunistic contacts and transmissions of road communications infrastructure to enable content retrievals in vehicles [10], [11], [12].

A. V2V Based System

Nandan *et al.* introduce the first V2V based content distribution protocol, namely SPAWN (swarming protocol for vehicular ad-hoc networks) [8], to enable the cooperative content retrieval and sharing among vehicles. In SPAWN, a file is first chopped into multiple pieces and then swapped among vehicles in a BitTorrent style to facilitate the collaborative download. Within the similar framework, Lee *et al.* propose CodeTorrent [13] which deploys the network coding to maximize the mutual differences of content pieces stored in the nearby vehicles, and accordingly reduces the search delay and coordinations of piece transmissions. Unlike SPAWN and CodeTorrent, our work mainly focuses on the design of the infrastructure and development of distributed content replication protocols. As the infrastructure based content distribution serves as a complement to the V2V content distribution, our proposal can work supplementally to [8], [9], [13].

Li *et al.* propose CodeOn [14] for efficient content distribution over vehicular networks in a highway. Yan *et al.* [15] develop an analytical model to evaluate the multi-hop transmission rate of the network coding based content distribution in the highway. Ye *et al.* [16] also investigate on the highway content distribution using network coding and develop an analytical model to evaluate the completion probability of content dissemination in the Rayleigh fading channel. Zhang *et al.* [17] develop a platoon-based content distribution protocol which optimally replicates content in a vehicle platoon based on the diverse mobilities of platoon members. In contrast to CodeOn, our work considers the content distribution in urban areas. Unlike that on highways, the V2V communication in urban tends to be much more dynamic due to the complicated street layout and diverse mobilities of vehicles. Moreover, the V2V communication in the urban has much lower capacity and smaller coverage due to the intense interferences within densely located nodes and shadowing and fading effects caused by complex building environments. In this case, we argue that it is desirable to explore the infrastructure for content distribution. However, the approach to deploy distributed infrastructure as proposed in this paper can also be applied in the highway vehicular networks.

Acer *et al.* [18] propose a V2V-based content distribution network to deliver the non-real time content information in the metropolitan area using the bus network. By exploring the stable bus schedule, predictable bus mobility and temporary storage at bus stops, a routing protocol is proposed which takes the randomness of rod traffic into consideration to deliver a single-copy file from the source to destination. Unlike [18], this paper targets to multicast a variety of files to a vast of vehicles. However, it is interesting to combine the bus network with our proposal by deploying the RSBs at bus stops and exploring the predictable bus mobility for content disseminations.

B. Vehicle-to-Infrastructure Based System

Zhang *et al.* [11] develop a scheduling algorithm at distributed RSUs to manage the V2R accesses of vehicles for service differential content distribution. Nandan *et al.* propose AdTorrent [19] to facilitate the distribution of advertising contents pertaining to a local area. In AdTorrent, static wireless digital billboards are deployed on the roadside which continually push the advertising contents, *e.g.*, hotel virtual tours, movie trailers, *etc.*, to the vehicles in proximity. Among vehicles, the advertising contents are then swapped in a BitTorrent style similar to SPAWN. [11] focuses the content schedule and service provision at a single RSU, and [19] investigates on the content distribution over a small region without considering the collaborative caching between wireless digital billboards and vehicles. In contrast, we target to support the content distribution infrastructure over a large region with a large-scale node population. To do so, it is key to intelligently and fully utilize the buffer resource of the distributed infrastructure devices.

Trullols-Cruces *et al.* [12] explore opportunistic contacts and cooperative download among vehicles to enhance the content delivery rate. Specifically, to distribute a file to the receiver, multiple relay vehicles are selected to carry the content file from roadside gateways to the in-motion receiver. This is enabled by the analysis of node mobility and road traffic. Similar to [18], [12] also focuses on the single-copy file delivery whereas our work focuses on multicasting files to a group of interested users.

The paper by Huang *et al.* [20] is relevant to our work, in which a buffer storage infrastructure is proposed to enable the content retrieval of vehicles in a region. Without involving the V2V communications, [20] seeks the optimal content replications along the path of vehicles to maximize the delivery ratio of contents for each drive. On achieving this goal, it is assumed that the path information of vehicles is available as an input to a centralized system for a suboptimal solution. Our work differs from [20] in three aspects. Firstly, [20] assumes that the mobility trajectories of vehicles are given and used for determining the content replications accordingly. Our work, however, relaxes the assumption of given mobility trajectories, but relies on random mobility of vehicles with given statistics of the connection time to RSBs. Secondly, [20] targets to maximize the number of content files that can be distributed to vehicles along their trajectories. Our work targets to maximize the global network utility, and towards this goal, we develop mathematical framework to evaluate the file download delay. Lastly, [20] relies on the infrastructure for content distribution without the assistance of V2V communications. Our work allows the collaborative file download among vehicles using the V2V communications.

III. System Model and Problem Formulation

In this section, we present the system model, including the RSB modelling, mobility of vehicles and the utility function of vehicular users. Based on the system model, we formulate the network design as an optimization problem. The main notations used are summarized in Table I.

A. System Model

1) Model of RSBs: We consider the city as a bounded region where a set $\mathbf{R}$ of RSBs are randomly deployed. Note that with different building environments and diverse communication capabilities, RSBs at different locations would have different radio coverage. Within their communication coverage, we consider RSBs to have the same data transmission rate to vehicle nodes, denoted by C_{V2R}. In this work, we allow vehicles to communicate with each other to cooperatively disseminate the downloaded contents to each other. Let C_{V2V} denote the data transmission rate of V2V communications. Each vehicle is equipped with a single-radio transceiver and communicate to only one node at each time, which is same as [21]. We make $C_{\mathsf{V2R}} > C_{\mathsf{V2V}}$, and vehicles prefer to downloading from RSBs if RSB connections are available. This is a working assumption as RSBs tend to have higher transmission rate than V2V communications due to the ample power energy and better channel conditions when mounted high [5].

2) Model of Node Mobility: The mobility of each vehicle node is represented by an *on-off* process based on its connectivity to RSBs: a vehicle node is in state 0 if it is outside the coverage of any RSB; otherwise, it is in state 1. Due to the random radio coverage and deployment locations of RSBs, we model the sojourn time of vehicles in state 1 and state 0 by the unpredictable, memoryless and continuous-time setting, following an exponential distribution with the mean value $1/\lambda$ and $1/\mu$, respectively.

3) Model of Files: Let $\mathbf{F}$ denote the integrated set of content files available for download in the region of interest. Throughout the work, each RSB is assumed to be manipulated

TABLE I
SUMMARY OF NOTATIONS

Notations	Description
$\mathbf{R}$	Set of RSBs in the region of interest
$\mathbf{F}$	Set of files published for download in the region of interest
$\mathbf{P}$	Popularity profile of files, where each element p_i, $i \in \mathbf{F}$, represents the probability that a user subscribes to download file i upon each download request
$\mathbf{A}$	Availability profile of files, where each element a_i, $i \in \mathbf{F}$, represents the portion of users which have file i stored in the local buffer
$\mathbf{B}$	Caching profile of files, where each element b_i, $i \in \mathbf{F}$, represents the probability that file i is stored in a RSB
C_{V2R}	Communication data rate between RSBs and vehicles
C_{V2V}	Communication data rate of V2V communications
B_{R}	Buffer size of a RSB
$U(\cdot)$	Utility function of vehicular users
$\mathcal{U}$	Global network utility (overall performance of network to optimize)
r	Download throughput of vehicles when vehicles are outside the communication of RSBs
R	Download throughput of vehicles when vehicles are inside the communication of RSBs
n	Number of vehicles which are able to transmit to the tagged node, or contend the channel for transmission with the tagged node
κ_i	Number of blocks in file i
τ_i	Mean download delay of file i
$1/\lambda$	Mean sojourn time of vehicles inside the communication range of RSBs
$1/\mu$	Mean sojourn time of vehicles outside the communication range of RSBs
$1/\delta$	Mean file block download time of vehicles outside the communication range of RSBs
$1/\gamma$	Mean file block download time of vehicles inside the communication range of RSBs
$\Gamma(m,k)$	Mean first passage time of Markov process from state (m,k) to state $(\cdot,\kappa_i)$
$\lvert\cdot\rvert$	Cardinality of set
$\langle\cdot\rangle$	Mean value of a random variable
$\mathrm{Var}(\cdot)$	Variance of a random variable

by a distinct owner; the owner uploads contents to its RSB at periodic intervals following the exponential distribution with the mean Δ. RSBs have homogenous buffer size[1] which is denoted by L. When the buffer of RSBs overflows with excessive file uploading from vehicles, the oldest files stored in the RSB will be evicted[2].

Throughout the work, we focus on the design of RSBs and assume that the buffer management at vehicle nodes are predefined and out of the control. In specific, the vehicles could have heterogenous and limited sized buffer storage, and randomly select files to evict if their buffer overflows. The pattern of V2V content swap is also predefined which could follow existing schemes, such as SPAWN [8].

With new files being continually published at distributed RSBs and old files being evicted from the network, $\mathbf{F}$ dynamically changes over time. In the network, each file is characterized by a three-tuple, including file blocks, popularity and availability, defined as follows.

a) File Blocks: Each content file in the network is divided into multiple non-overlapping file blocks for delivery. In order to finish downloading a file, a vehicle node must collect all blocks of the requested file from either RSBs or other vehicles with the file stored. A vehicle node can only redistribute a file to the others after it has the entire file downloaded and recovered[3]. Let κ_i denote the number of blocks of file i, where $i \in \mathbf{F}$. For ease of analysis, we assume that all the blocks of files have equal size. With files having different numbers of blocks, they are heterogenous in size. For computation simplicity, L, C_{V2R} and C_{V2V} are normalized by the block size.

b) File Popularity and Availability: Besides the number of file blocks, each content file in the network is characterized by another two parameters, namely popularity and availability.

Definition 1: The popularity p_i of file i, where $i \in \mathbf{F}$, represents the probability that a vehicle subscribes to download file i upon each download request which it issues. The *popularity profile* of $\mathbf{F}$ is a $1 \times |\mathbf{F}|$ probability vector $\mathbf{P} = \{p_i; i \in \mathbf{F}\}$, where $|\cdot|$ indicates the cardinality of set.

Definition 2: The availability a_i of file i, where $i \in \mathbf{F}$, represents the probability that a randomly selected vehicle has file i cached in its buffer. The *availability profile* of $\mathbf{F}$ is a $1 \times |\mathbf{F}|$ probability vector denoted by $\mathbf{A} = \{a_i; i \in \mathbf{F}\}$.

Note that since $\mathbf{F}$ is dynamically changing over time, the popularity profile $\mathbf{P}$ and availability profile $\mathbf{A}$ are also varying

[1] In practice, the RSBs would be produced by the same vendor with equal buffer size.

[2] It is interesting to investigate on the impacts of different buffer management schemes, *e.g.*, least frequently used (LFU) and least recently used (LRU) on the download performance, which however is out the scope of this work.

[3] It can be extended by allowing vehicles to redistribute file blocks as long as certain blocks are downloaded in entirety. We study the simplest case and leave the extension for future works.

over time. In this case, RSBs stochastically select files in $\mathbf{F}$ to cache in their buffer following the caching profile defined below.

Definition 3: Let b_i denote the probability that a randomly selected RSB has file i stored in its buffer. The *caching profile* of $\mathbf{F}$ is a $1 \times |\mathbf{F}|$ probability vector denoted by $\mathbf{B} = \{b_i; i \in \mathbf{F}\}$.

4) Mean Download Delay of Files: The performance of the network is characterized by the mean download delay of files. Let τ_i denote the mean download delay of file i which starts when a download request of file i is issued by a vehicle until the the vehicle finishes downloading all the blocks of file i. Given the distribution of RSBs and density of vehicle nodes in the region of interest, the download delay τ_i is dependent on the availability of file i at RSBs, represented by b_i, and vehicles, represented by a_i.

B. Network Utility Function

For each file i, we assume that there is an underlying utility function $U_i(\tau_i)$ that specifies the satisfaction of vehicular users on the download of file i provided the download delay τ_i. Moreover, it is nature to assume that $U(\tau_i)$ is a monotonically decreasing function of τ_i, *i.e.*, reducing the download delay τ_i would monotonically increase the user's utility of file i.

The proposed infrastructure is designed to maximize a global network utility function $\mathcal{U}$ which represents the integrated utilities of vehicles. In general cases, it can be expressed as a weighted sum of individual user utilities over all files, mathematically,

$$\mathcal{U} = \sum_{i \in \mathbf{F}} w_i U(\tau_i), \tag{1}$$

where $w_i, i \in \mathbf{F}$, is a given positive weight. With different concerns, the network utility can be adapted to achieve different design goals, as following examples:

1) User-centric Content Distribution: In this scenario, by tuning the weighting factor of each file equal to the corresponding file popularity, the proposed infrastructure targets to optimize the user's download experience by maximizing the average user satisfaction on the file dissemination. Mathematically, the network utility is given as

$$\mathcal{U} = \sum_{i \in \mathbf{F}} p_i U(\tau_i). \tag{2}$$

2) Content-centric Content Distribution: The weighting factor w_i can be set to a predefined value which reflects the importance of file i. For example, breaking news, important software update, *etc.*, can be assigned with the large weighting factors and accordingly attain high priorities to be stored in RSBs. This ensures those important files to be vastly stored and ubiquitously available.

3) Cost-centric Content Distribution: A practical concern of the proposed infrastructure is the hardware cost of RSBs. With larger buffer storage of RSBs, more files can be cached in each RSB, rendering reduced download delay to users; nevertheless, it increases the cost of RSB hardware accordingly. Motivated by this concern, the network utility can be modified by introducing the cost function in (1) to strike a trade-off between the network performance and investment cost, as

$$\mathcal{U} = \sum_{i \in \mathbf{F}} w_i U(\tau_i) - \mathbf{C}(L), \tag{3}$$

where $\mathbf{C}(L)$ represents the hardware cost of RSBs which is a non-decreasing function of the buffer size L. In practice, $\mathbf{C}(L)$ ca'n be evaluated by $\mathbf{C}\left(\sum_{i \in \mathbf{F}} b_i \kappa_i\right)$ instead, where $\sum_{i \in \mathbf{F}} b_i \kappa_i$ represents the mean usage of RSB buffer storage. As such, the three designs of the network as aforementioned can be solved using the unified formulation as described below.

C. Problem Formulation

Given the network model introduced above, each RSB distributedly determine the optimal caching profile $\mathbf{B}$ of files to attain the maximal network utility $\mathcal{U}$, mathematically,

$$\begin{array}{ll} maximize & \mathcal{U} \\ subject\ to: & \Pr(X \geq L) \leq \varepsilon, \\ & b_i \in [0,1], \qquad i \in \mathbf{F}, \end{array} \tag{4}$$

where X denotes the usage of the RSB buffer storage at any time, and $0 < \varepsilon << 1$ is a predefined constant. The constraint of (4) specifies that the overflow probability of each RSB should be no larger than ε.

Lemma 1: Let X_i, $i \in \mathbf{F}$, be an independent binary random variable with the probability mass function

$$\Pr(X_i = 1) = b_i, \quad \Pr(X_i = 0) = 1 - b_i,$$

which denotes whether file i is stored in a RSB or not. For $X = \sum_{i \in \mathbf{F}} X_i \kappa_i$ with $\kappa_i > 0$, which denotes the usage of RSB buffer storage, we have $E(X) = \sum_{i \in \mathbf{F}} b_i \kappa_i$. By denoting $v = \sum_{i \in \mathbf{F}} b_i \kappa_i^2$, we have

$$\Pr(X \geq E(X) + \psi) \leq \exp\left(-\frac{\psi^2}{2(v + \kappa\psi/3)}\right) \tag{5}$$

where $\kappa = \max\{\kappa_i; i \in \mathbf{F}\}$.

Proof: Refer to [22] (pp.25). ■

Theorem 1: Given the network modeling, the constraint of (4) is achieved when

$$E(X) \leq L - \kappa \frac{2}{3} \log \epsilon - \sqrt{\kappa^2 \frac{4}{9} \log^2 \epsilon - 2\kappa L \log \epsilon}. \tag{6}$$

Proof: Refer to Appendix A. ■

Denote by $\mathcal{L} = L - \kappa \frac{2}{3} \log \epsilon - \sqrt{\kappa^2 \frac{4}{9} \log^2 \epsilon - 2\kappa L \log \epsilon}$. With Theorem 1, (4) can be modified as

$$\begin{array}{lll} \mathsf{OPT} & maximize & \mathcal{U} \\ & subject\ to: & \sum_{i \in \mathbf{F}} b_i \kappa_i \leq \mathcal{L}, \\ & & b_i \in [0,1], \qquad i \in \mathbf{F}. \end{array} \tag{7}$$

In (7), with $\mathbf{A}$ and $\mathbf{P}$ provided, tuning the caching profile $\mathbf{B}$, *i.e.*, content replications in RSBs, will adapt the download delay τ_i of each file i and accordingly lead to different network utility $\mathcal{U}$. In this work, our goal is to determine the solution of (7) in a distributed manner.

D. Evaluation of File Download Delay

To solve (7), the foremost issue is to identify the quantitative relation between the file download delay and the caching

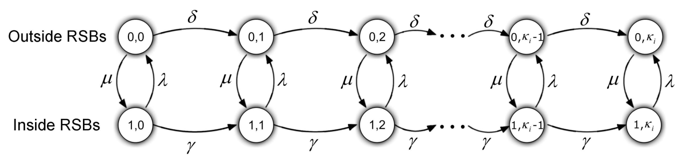


Fig. 2. State space and transitions of the two-dimensional Markov process

profile $\mathbf{B}$. To this end, we randomly select a vehicle node from the network (referred to as the tagged node) and evaluate its download delay of file i. Specifically, based on the system model described in Section III-A, we represent the tagged vehicle node by a two-dimensional Markov process $(M_i(t), K_i(t))$. Here, $M_i(t) \in \{0,1\}$ represents the mobility of the vehicle node according to the on-off model described in Subsection III-A, and $K_i(t) \in \{0,1,...,\kappa_i\}$ represents the number of file blocks that the tagged node has downloaded until time t. Fig. 2 shows the state space of the Markov process and all the non-null transitions. In what follows, we evaluate the transition rates of the Markov process according to the locations of the tagged node.

1) Tagged Vehicle Outside the Coverage of RSBs: When the tagged node is o utside the coverage of any RSBs, it can only download from nearby vehicles using the V2V communications; at each time, we refer to the set of vehicles which are within the communication range of the tagged node as the neighbor nodes. Let n denote the number of neighbor nodes of the tagged node; n is a random variable, and let $\langle n \rangle$ and $\mathrm{Var}(n)$ denote its mean and variance, respectively. Assuming that the vehicular network has an ideal MAC where the channel airtime is fairly shared among the nearby vehicles, the throughput of the tagged node using the V2V communication is a function of n as,

$$r = \frac{C_{\mathsf{V2V}}}{n+1} Q_i(n), \tag{8}$$

where $Q_i(n) = 1 - (1-a_i)^n$, representing the probability that at least one neighbor node of the tagged node has file i stored in its buffer, or equivalently, the probability that the tagged node can retrieve file i from its neighbor nodes. $(n+1)$ in (8) represents the number of vehicles fairly sharing the channel.

Let δ denote the transition rate from state $(0, K_i(t))$ to state $(0, K_i(t)+1)$, where $K_i(t) \in \{0,1,...,\kappa_i - 1\}$, as shown in Fig. 2. Assuming that the download time of one block using the V2V communication follows the exponential distribution, δ is equal to the mean V2V communication throughput $\langle r \rangle$ where r is specified in (8). Taking the expectation on n in both sides of (8), we approximate $\langle r \rangle$ using the second order Taylor series approximation as shown in Lemma 2.

Lemma 2: With the second order Taylor approximation, we have

$$\langle r \rangle \approx r|_{\langle n \rangle} + \frac{1}{2}\mathrm{Var}(n) \left. \frac{d^2 r}{dn^2} \right|_{\langle n \rangle}. \tag{9}$$

Proof: Refer to Appendix B. ■

2) Tagged Vehicle Inside the Coverage of RSBs: In this case, the tagged node can download the demanded blocks from either neighbor vehicles or the RSB. We assume that the tagged vehicle would select to download from RSBs with high priority, if the connected RSBs have the desired file stored; otherwise, it would download from neighbor vehicle nodes. This is because that RSBs have the greater communication capacity than vehicles [5]. In this scenario, given that file i is stored at the RSB with probability b_i, the download throughput of the tagged vehicle in this scenario is

$$R = b_i \frac{C_{\mathsf{V2R}}}{n+1} + (1-b_i)\, r. \tag{10}$$

The first component on the right-hand-side of (10) represents the download rate from the RSB with the ideal MAC applied, and the second component on the right-hand-side of (10) represents the download rate using the V2V communications given that with probability $(1-b_i)$ that the RSB does not have the desired file i stored.

Let γ denote the transition rate from the state $(1, K_i(t))$ to the state $(1, K_i(t)+1)$, where $K_i(t) \in \{0,1,...,\kappa_i - 1\}$, as shown in Fig. 2. Similar to the previous case, we assume that the download time of one block inside the RSB follows the exponential distribution. Therefore, we have γ equal to $\langle R \rangle$ with R shown in (10). $\langle R \rangle$ can be approximated with the second order Taylor approximation as in Lemma 3.

Lemma 3: With the second order Taylor approximation, we have

$$\langle R \rangle \approx b_i \Phi + (1-b_2)\langle r \rangle, \tag{11}$$

where $\Phi = C_{\mathsf{V2R}} \left(\frac{1}{\langle n \rangle + 1} + \frac{\mathrm{Var}(n)}{(\langle n \rangle + 1)^3} \right)$

Proof: Refer to Appendix C. ■

3) Mean File Download Delay: We evaluate the average file download delay by the mean first passage time stating from the state $K_i(0) = 0$, *i.e.*, no blocks are downloaded, until the state $K_i(t) = \kappa_i$, *i.e.*, the tagged node collects all the desired file blocks. Let $\Gamma(m,k)$ denote the first passage time starting when the vehicle is in state (m,k) until all κ_i blocks are downloaded, mathematically,

$$\Gamma_i(m,k) = \min\{t > 0 |\, M_i(0) = m, K_i(0) = k \text{ and } K_i(t) = \kappa_i\}.$$

The mean download delay of file i is thus

$$\tau_i = \frac{1}{\lambda + \mu} \left(\lambda \Gamma_i(0,0) + \mu \Gamma_i(1,0) \right), \tag{12}$$

where $\frac{\lambda}{\lambda+\mu}$ and $\frac{\mu}{\lambda+\mu}$ are the limiting probabilities that the tagged node is outside and inside the coverage of RSBs, respectively, when the tagged node initiates the download subscription of file i. The expression of τ_i is given in Theorem 2.

Theorem 2: The average download delay of file i is

$$\tau_i \approx \frac{b_i(\Phi - \delta)\lambda + \delta(\lambda+\mu) + \kappa_i(\lambda+\mu)^2}{b_i(\Phi-\delta)\mu(\lambda+\mu) + \delta(\lambda+\mu)^2}. \tag{13}$$

Proof: Refer to Appendix D. ■

Corollary 1: The download delay τ_i is a monotonic non-increasing convex function of b_i if $\frac{\kappa_i}{\delta} \geq \frac{1}{\mu} - \frac{2}{\lambda+\mu}$.

Proof: Refer to Appendix E. ■

IV. Protocol Description

By substituting (13) into (1), we are now ready to derive the solution of (7).

A. Sufficient Conditions for a Concave Network utility Function

We make two assumptions as follows:

- ▷ $\frac{\kappa_i}{\delta} \geq \frac{1}{\mu}$, for all i, which implies that the average download time of a file when vehicles are outside the RSBs (evaluated as κ_i/δ) should be no smaller than the average sojourn time of vehicle nodes outside the RSBs (evaluated as $1/\mu$). Otherwise, the assistance of RSBs is negligible as the desired file can be downloaded easily through V2V communications only before vehicles entering into the coverage of any RSBs.
- ▷ $U(\tau_i)$ is a non-increasing, twice differentiable concave function of τ_i. As an example to explain the methodology, we adopt

$$U(\tau_i) = -\tau_i \quad \text{and} \quad w_i = p_i,\ i \in \mathbf{F}, \tag{14}$$

for the simplicity.

According to Corollary (1) and Proposition (1), the network utility $\mathcal{U}$ is a concave function of b_i, and accordingly, the network utility maximization problem in (7) is a convex optimization problem.

Proposition 1: If $U(\tau_i)$ is a non-increasing, twice differentiable concave function of τ_i, then $U(\tau_i)$ and network utility $\mathcal{U}$ are concave functions of b_i.

Proof: Refer to Appendix F. ■

B. Global Optimal Solution to (7)

Let $\mathbf{B}^*$ denote the optimal caching profile. By examining (7) with the Karush–Kuhn–Tucker (KKT) conditions as shown in Appendix G, we have $\mathbf{B}^*$ as in (15).

Then given the availability profile $\mathbf{A}$ and popularity profile $\mathbf{P}$, each RSB should select content file i to cache with probability b_i^*. We refer to this scheme as the global optimal replication in RSBs.

The global optimal replication provides the optimal solution to (7). However, note that both $\mathbf{A}$ and $\mathbf{P}$ are system-wide parameters which relate to the file information across the whole network. They are not available to individual RSBs or vehicle nodes when the network size is large. Therefore, the global optimal replication scheme is not practical for large-scale real-world deployment. Nevertheless, the global optimal replication provides a benchmark for performance comparison with other replication schemes. In what follows, we propose a decentralized algorithm to determine the content replication at RSBs.

C. Distributed Content Replication

In this part, we design a distributed algorithm to enable RSBs to select the appropriate files to store according to (7) in a fully distributed manner. To achieve this goal, we approximate b_i^* by b_i^{d} as

$$b_i^{\mathsf{d}} = \frac{\mathcal{L}\sqrt{p_i\left[\kappa_i\mu(\lambda+\mu)+\delta(\mu-\lambda)\right]}}{\sum_{j\in\mathbf{F}}\kappa_j\sqrt{p_j\left[\kappa_j\mu(\lambda+\mu)+\delta(\mu-\lambda)\right]}}. \tag{16}$$

This can greatly simplify the algorithm design with modest performance degradation as verified by simulations.

To help RSBs distributedly select file i with the probability b_i^{d} from the network, we adopt a random walk based algorithm over a file graph as follows:

1) File Graph: The file graph refers to as a graph connecting all the files stored in distributed vehicles. As an example shown in Fig. 3, each vertex in the graph represents a file stored in a vehicle node. Additionally, each vehicle has an anchor file, *e.g.*, file j, which is selected from the locally stored files in vehicles and has the largest value of $\sqrt{p_j\Phi\left[\kappa_j\mu(\lambda+\mu)+\delta(\mu-\lambda)\right]}$ among the buffered files. Each vehicle node periodically broadcasts its anchor file information, including the availability a_j and popularity p_j, to the neighbor vehicles. How to measure the availability and download demand of files will be discribed in Section IV-C2a. In the file graph, all files stored in the same vehicle node are fully connected, and the anchor files among neighboring vehicles are fully connected, as shown in Fig. 3. Therefore, the file graph has a two-tier architecture where the top tier connects the anchor files of vehicles and the underlying tier connects all the files inside a vehicle to its anchor file.

2) Random Walk Based File Selection: The file selection is realized by a random walk algorithm over the file graph as described in Algorithm 1. Specifically, to determine the files stored in RSBs, an RSB first issues a number η of random walkers to separate vehicles in the communication range. Each vehicle which receives a walker will then initiate the random walk process starting from its anchor file. The walker is forwarded stochastically on the file graph from one vertex (file) to another vertex (file) following the Metropolis-Hasting algorithm; the derivation of transition probabilities in the random walk is given in Appendix H. Once the walker is forwarded to the anchor file, it may be relayed to other anchor files stored in different vehicles. In this case, the walker is forwarded to other vehicles and proceeds the random walk algorithm. After being relayed for Time-To-Live (TTL) hops among files on the file graph including self-loops, the walker stops at a file which is then selected to be uploaded to RSBs.

In order to compute the transition probability of the walker, each vehicle needs to know the availability and popularity of the files stored in its buffer. In the proposed infrastructure, we enable vehicles to distributedly measure these parameters as follows:

a) Measurement of File Availability: The availability a_i of file i is only measured by the vehicles which needs to download file i. As each vehicle interested in file i continually

$$b_i^* = \frac{\sqrt{p_i\left[\kappa_i\mu(\lambda+\mu)+\delta(\mu-\lambda)\right]}}{\sum_{j\in\mathbf{F}}\kappa_j\sqrt{p_j\left[\kappa_j\mu(\lambda+\mu)+\delta(\mu-\lambda)\right]}}\left(\mathcal{L}+\frac{\delta(\lambda+\mu)}{\mu(\Phi-\delta)}\sum_{j\in\mathbf{F}}\kappa_j\right)-\frac{\delta(\lambda+\mu)}{\mu(\Phi-\delta)},\quad b_i^*\in\mathbf{B}^*. \tag{15}$$

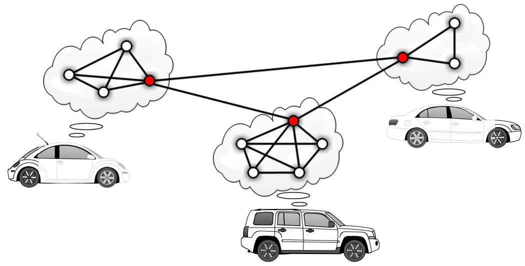


Fig. 3. File graph in the vehicular network

Algorithm 1: Random walk algorithm starting from file x

```
/* m: current file with walker                    */
/* h: hop account                                 */
/* p: random number                               */
/* P_mn: transition probability from
   file m to file n shown in (H-3) of
   Appendix H                                     */
begin
  Initialization: m ← x; h ← 0; p ← 0;
  while h < TTL do
    p ← random number in [0, 1] ;
    foreach file n (n ≠ m) connected to file m in the
    file graph do
      if p ≤ P_mn then
        m ← n;
        quit the foreach loop;
      else
        p ← p − P_mn;
    h ← h + 1;
Result: File m
```

issues the download requests to its neighboring vehicles, it can estimate the file availability a_i based on the replies with $\frac{\text{No. of vehicles having file } i \text{ stored}}{\text{Overall No. of vehicles contacted}}$. Whenever the vehicle, *e.g.*, x, interested in file i meets another vehicle, *e.g.*, y, which has file i stored, vehicle x would inform the measurement of a_i to vehicle y piggybacked with the download request. As such, vehicle y would receive multiple measurements of a_i. For each new measurement received, it would incorporate it with the previous measurement using the moving average. Once vehicle x finishes downloading file i, it can use its measurement on a_i to evaluate the availability of file i.

b) Measurement of File Popularity: The download demand d_i of file i is measured by the vehicles which have file i stored. As those vehicles keep receiving download requests from others and a portion of the requests are for file i, d_i can be estimated based on this information as $\frac{\text{No. of vehicles requesting to download file } i}{\text{Overall No. of download requests received}}$.

It is important to note that each vehicle only needs to know the available and download demand of the files it stores. Therefore, the distributed measurement will not impose much workload on the message exchange. To improve the measurements of availability and download demand of files, we can also make use of the RSBs. In this case, RSBs would collect the measurements from different vehicles driving through, and average the measurements towards a more accurate estimation, then announce them to vehicles. There would be other methods for more accurate measurements based on the vehicular sensor networks [23], which is out of the scope of this work.

D. Protocol Description

This part describes the detailed protocol design and implementation of the proposed infrastructure. In the network, each RSB, *e.g.*, A, works in a fully distributed manner and conducts the following three operations:

1) File Publication: Whenever a new file is published at RSB A (uploaded by its owner), the RSB A issues η walkers to separate vehicles in its coverage. Each walker is relayed among files over the file graph embedded in the vehicular networks following Algorithm 1, and results in one file selected after the TTL hops. The vehicles with the selected files will then upload the files to the RSBs which they drive through. As such, RSBs are dynamically refreshed with new contents continuously uploaded; and this process is triggered by the publication of new files. The value of η will be discussed later. Note that in this phase, RSB A is only responsible to issue walkers to the vehicular network upon the publication of new files. The files selected by the walkers will be uploaded to RSBs in the communication range of the vehicles hosting the selected file, which may not be RSB A.

The value of η is set to make the overall number of files in the network stable. It is dependent on the rates at which new files are published to the network and the out-dated files are evicted from the network. Let $\mathcal{T}$ be the average life time of files in the network, where the life time represents the time duration that a file is stored in RSBs. Let θ be the average injection rate of new contents to the network at distributed RSBs. As each newly published file will initiate η walkers to the vehicular network and finally cause η files to be uploaded to RSBs from vehicles, the rate at which RSBs get new contents uploaded is in total $(1+\eta)\,\theta$. Let $\mathcal{N}$ denote the number of RSBs in the overall network. Mathematically, the rate at which the number of content files changes over time is

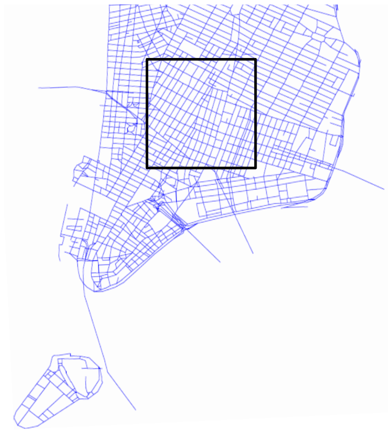
$$\frac{\partial\,|\mathbf{F}|}{\partial t} = (1+\eta)\,\theta\mathcal{N} - \frac{1}{\mathcal{T}}\mathcal{N}\mathcal{L}. \quad (17)$$

In the steady state with $\frac{\partial|\mathbf{F}|}{\partial t} = 0$, we have

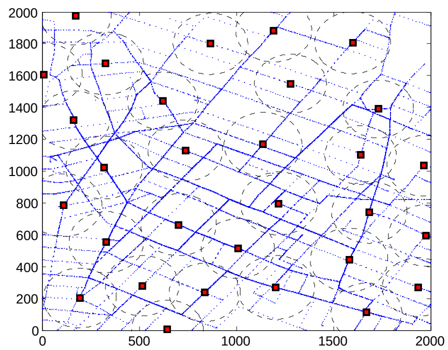
$$\eta = \frac{\mathcal{L}}{\mathcal{T}\theta} - 1. \quad (18)$$

To compute η with (18), we assume that θ and $\mathcal{T}$ are known through measurements at different RSBs distributively based on the history of file storage in RSBs. RSBs can also exchange the measurements among each other to improve the accuracy with the assistance of vehicles.

2) Retrieve Files from Vehicles: Whenever a vehicle with a selected file in the random walk algorithm comes into the coverage of RSB A, it will retrieve the file immediately from the vehicle. During this period, the channel of RSB A is used exclusively for the file retrieval. If there are multiple uploads simultaneously from different vehicles to RSB A, RSB A



(a) Street layout



(b) RSB distribution on the road

Fig. 4. Street layout based TIGER/Line Shapefiles [24], and RSB distribution in simulations

only processes one retrieval at one time until this retrieval completes. Once the selected file in a vehicle is uploaded, the vehicle will not upload this file to other RSBs unless this file is selected again in the random walk algorithm. In case that a vehicle moves out of the coverage of RSB A before it accomplishes the retrieval, RSB i would proceed the file retrieval again from other driving through vehicles which has the unfinished file stored. If its buffer is full, RSB A depletes the buffer by deleting the file which has been stored for the longest time.

3) Upload File to Vehicles: In the idle period of RSB i when it does not need to issue walkers to the network or retrieve files from vehicles, it uploads the cached file to the driving through vehicles upon their requests.

Each RSB in the network thus works in the three modes in a fully distributed manner. In what follows, we evaluate the performance of the proposed infrastructure compared to the centralized content replication.

V. SIMULATIONS

This section evaluates the performance of the proposed infrastructure using simulations based on a discrete-event simulator coded in C++.

A. Simulation Setup

Our simulation is carried out over a 1.5 km$\times$$1.5$ km regional road map on the Manhattan island with the contour of the street layout plotted in Fig. 4. Each road segment in Fig. 4(a) is of two lanes with the bidirectional vehicle traffic. Compromised to the complexity of simulations, we select a bounded region on the map for our simulations, as shown in Fig. 4(b). There are totally 29 RSBs deployed in the region with the communication range uniformly distributed within the range $[180, 200]$ meters. For each simulation run, 300 vehicles are involved in the content distribution. The mobilities of vehicles are generated by VANETMobisim [25], in which the destination of each trip is randomly selected, and the velocity of each vehicle is controlled no larger than 60 km/s and adapted by the IDM-LC (Intelligent Driver Model with Lane Changes) mode. The coverage of V2V communications is set to be 150 meters. With this configuration, we have $\lambda = 29.46$s, $\mu = 12.19$s, $\langle n \rangle = 4.02$ and Var$(n) = 6.18$. In each simulation run, 200 files are initially available for download in the network, which are randomly stored in RSBs and vehicles. The file size is accounted in the unit of blocks and uniformly distributed within $[40, 100]$ blocks. Unless mentioned otherwise, all RSBs have the equal buffer storage to cache 3×10^3 file blocks, *i.e.*, $3 \times 10^3/100 = 30$ files at most. Vehicles have equal buffer to cache 1×10^3 file blocks, *i.e.*, 10 files at most. The download capacity of the vehicle to RSB communication, C_{V2R}, is 50 blocks/sec and that of the V2V communication, C_{V2V}, is 20 blocks/sec. Each vehicle can communicate with one other network component at most, and the parallel communication sessions are scheduled through the ideal MAC.

B. Verification of the Analysis

As the proposed protocol is based on the evaluation of file download delay, in the first experiment, we verify the accuracy of (9), (16) in evaluating the mean download rates $\langle r \rangle$ and $\langle R \rangle$, when vehicles are inside and outside RSBs, respectively, and the accuracy of (13) in evaluating the mean download delay of files. To this end, we carry out the Mento Carlo simulations by investigating on the download performance of a tagged vehicle. We make the tagged vehicle subscribe to download a file, referred to as file i in this section, of file size to be 100 blocks and report the averaged results over 5000 simulation runs.

Fig. 5 shows the values of $\langle r \rangle$ and $\langle R \rangle$ as a function of b_i when a_i is 0.1. As we can see from the figure, $\langle r \rangle$ remains the same with different b_i, and $\langle R \rangle$ increases linearly with b_i. The analyses in (9), (16) match the simulations well. Fig. 6 shows the mean download delay of the file with different b_i. As we can see, when b_i increases, the download delay τ_i reduces dramatically which can be characterized by (13). Moreover, τ_i is a convex function of b which validates Corollary 1. In addition, when a_i changes from 0.1 to 0.4, the download delay τ_i reduces significantly, as in this case more vehicles on the road have file i stored in their local buffer and therefore the tagged node can finish downloading faster.

Fig. 7 shows the values of $\langle r \rangle$ and $\langle R \rangle$ as a function of a_i when b_i is 0.1. As we can see, by increasing a_i, both $\langle r \rangle$ and $\langle R \rangle$ increase with a constant gap between the two curves which can be characterized by (16). As shown in Fig. 8 and indicated by (13), the mean download delay of file i reduces when a_i increases.

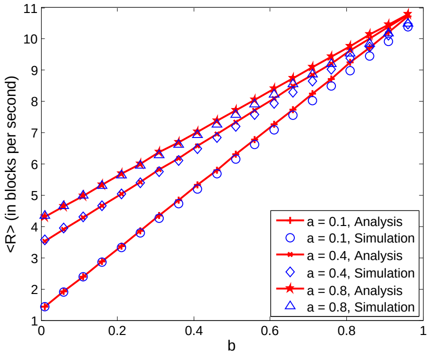


Fig. 5. Mean download rate with different values of file cache probability b

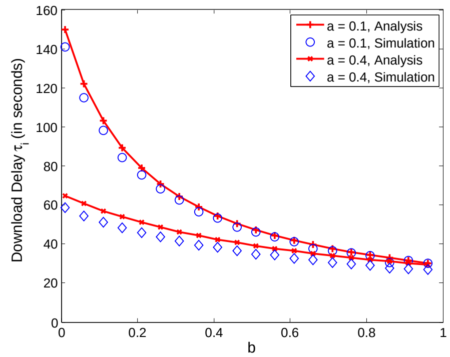


Fig. 6. Mean file download delay with different values of file cache probability b

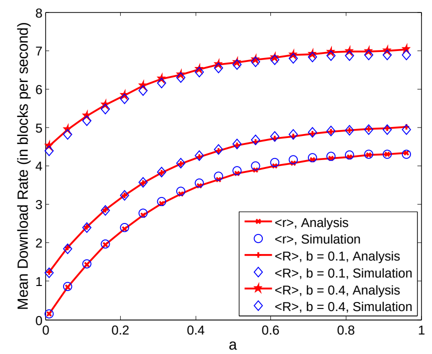


Fig. 7. Mean download rate with different values of file availability a

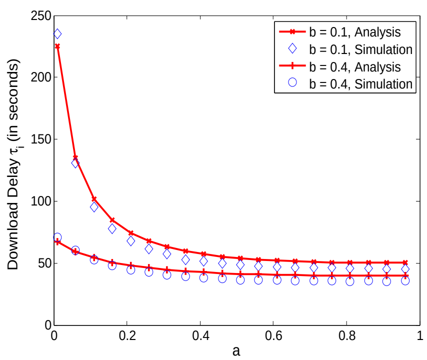


Fig. 8. Mean file download delay with different values of file availability a

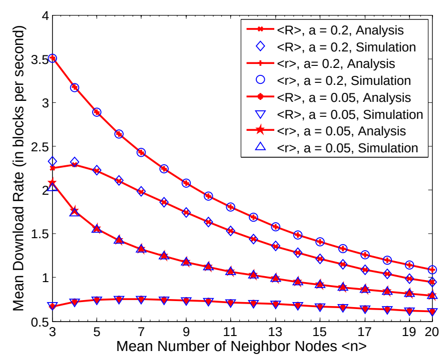


Fig. 9. Mean download rate with different values of $\langle n \rangle$

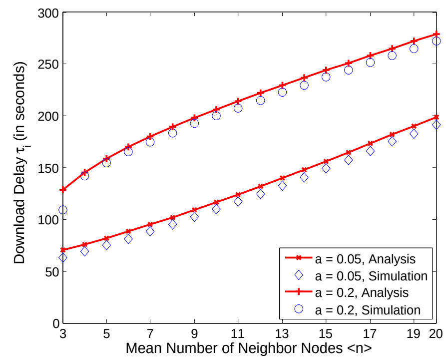


Fig. 10. Mean file download delay with different values of $\langle n \rangle$

Fig. 9 shows the values of $\langle r \rangle$ and $\langle R \rangle$ as a function of the mean number of neighbor vehicles, *i.e.*, $\langle n \rangle$, when b_i is 0.1. As we can see, by increasing $\langle n \rangle$, $\langle R \rangle$ reduces monotonically. This is because that with $\langle n \rangle$ increasing more vehicles share the capacity of RSBs and contend the channel with the tagged node. As in Fig. 9, in both cases when $a_i = 0.2$ and 0.05, respectively, $\langle r \rangle$ increases first when $\langle n \rangle$ increases and then reduces. This is because that when $\langle n \rangle$ increases, more neighbor vehicles may have the desired file i stored and upload the file to the tagged node. However, when $\langle n \rangle$ is large, indicating that more neighbor nodes are contending the channel with the tagged node, $\langle r \rangle$ reduces with $\langle n \rangle$ increasing. Fig. 10 shows the download delay of file i, τ_i, as a function of $\langle n \rangle$. As we can see, by increasing $\langle n \rangle$, τ_i increases. This is because that the download rate of the tagged node reduces as shown in Fig. 9.

C. Performance of Protocol

We simulate a dynamic network in which each RSB periodically publishes a new file to the network at the intervals following the exponential distribution with the mean of 60 seconds. The index of files increases linearly according to the publication time of the file in the network. Files have different popularity which follows the Zipf distribution; the popularity of the ith file in the network is as

$$p_i = \frac{1}{(\hat{i})^{\alpha}} / \sum_{j=1}^{|\mathbf{F}|} \frac{1}{j^{\alpha}}, \tag{19}$$

where α is a configurable parameter of the Zipf function. $\hat{i} = (i \bmod 500)$ where mod denotes the modulo operation. In this case, the popularity of files renews whenever 500 new files are published. The lifetime of each file is set to be 200 seconds which results in 5 walkers generated per RSB according to (18) when a new file is published. RSBs selectively retrieve files from vehicles based on the content replication scheme presented in Section IV. When the buffers of RSBs overflow, the file which has been stored for the longest time in the buffer is evicted. Vehicles select files to download based on the Zipf distribution as aforementioned. Once the buffer of vehicles is full, a randomly selected file is evicted to release the cache for new downloads.

We evaluate the utility function $\mathcal{U} = -\sum_{i \in \mathbf{F}} p_i \tau_i$ every 100 sections, which accounts for the summed download delay of files weighted by the file popularity within this period. We conduct 50 runs upon each simulation experiment and plot the mean result with the 95% confidence intervals.

Fig. 11 shows the comparison between the proposed random walk based content replication scheme and the global optimal and local greedy content replication schemes. The three schemes adopt the same content upload and download operations between vehicles and RSBs, except for the file selection strategies when individual RSBs retrieve files to store. Using the global optimal scheme, each RSB selects a file, *e.g.*, i, from the drive through vehicles to store with the probability b_i^* as shown in (15). Using the local greedy algorithm, each RSB selects a file with the largest value of

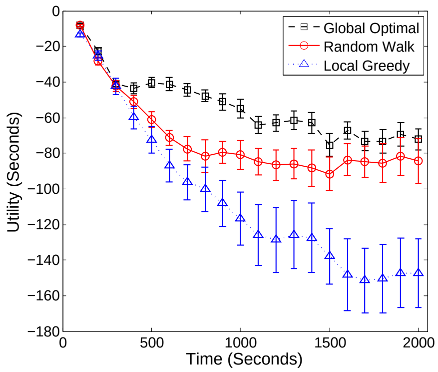


Fig. 11. Comparison between global optimal, random walk based and local greedy content replication schemes

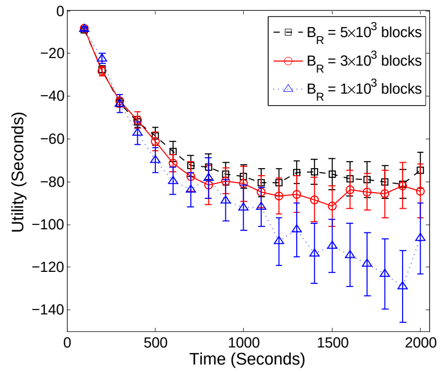


Fig. 12. Global network utility with different B_{R}

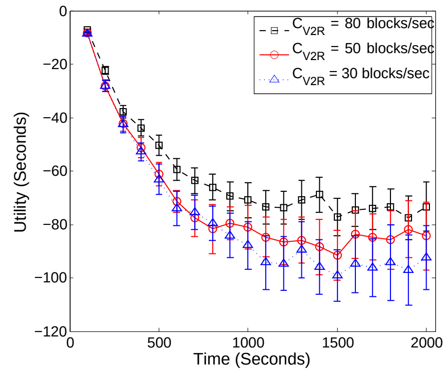


Fig. 13. Global network utility with different C_{V2R}

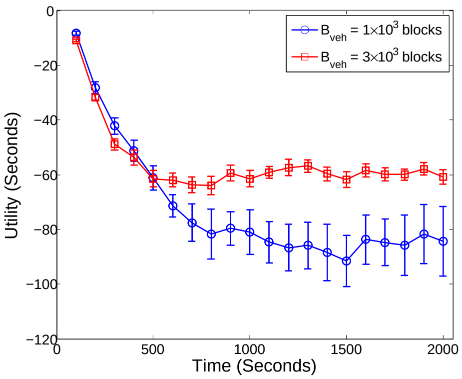


Fig. 14. Global network utility with different vehicular buffer B_{veh}

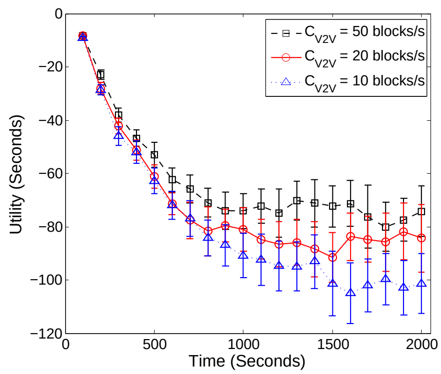


Fig. 15. Global network utility with different C_{V2V}

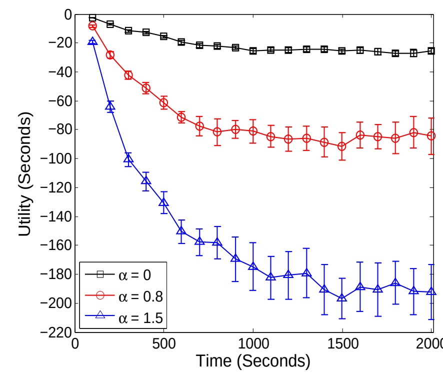


Fig. 16. Performance with different values of α

file popularity p_i to store. The strategy of file selection using the random walk based algorithm is discussed in Section IV. As we can see from Fig. 11, the global optimal scheme has the best performance, followed by the random walk algorithm. The local greedy algorithm has the worst performance for two reasons. Firstly, by selecting files with the local maximal file popularity to store, the local greedy scheme is myopic and cannot optimize the overall performance of the network. Secondly, without considering the storage of vehicles, using the local greedy scheme, RSBs may store files which have already been vastly stored in vehicles, and therefore, cannot be efficiently utilized to cooperate with the vehicular storage towards maximal social welfare. Notably, in all the schemes, the global network utility $\mathcal{U}$ reduces over time and finally approaches to a stabilized value. This is because that at each interval, $\mathcal{U}$ is evaluated by summing up the weighted download delays of files downloaded. Therefore, in the early period of simulations, only files with small delays are finished and accounted, leading to the small value of $\mathcal{U}$. Eventually, files with long download delays are accounted when they are finished, and the value of $\mathcal{U}$ becomes stable accordingly.

Figs. 12 and 13 show the global network utility with different buffer sizes and communication capacity of RSBs, respectively, with other parameters remaining the same. As we can see, when the buffer size or communication capacity increase, the global network utility increases, indicating smaller download delay of files. However, enhancing the buffer storage and communication capacity will lead to the increased physical cost of RSBs which discourages the large-scale deployment of RSBs.

Figs. 14 and 15 show the global network utility when increasing the buffer size and V2V capacity of vehicles, respectively, with other parameters remaining the same. As we can see, increasing the vehicles' buffer size or the capacity of V2V communications will significantly improve the network performance, as more bandwidth and storage resource are available in the network. However, in practice, the capacity and buffer storage contributed by vehicles are out of control. This therefore calls for an effective incentive mechanism to encourage contributions.

Fig. 16 shows the global network utility with different values of α in the Zipf function (19). $\alpha = 0$ indicates that all the files have the equal popularity; the larger α is, the faster that the popularity of files decreases when $\hat{i}$ increases. When α increases, the global network utility reduces significantly. This is because that there exists certain very popular files which are highly demanded. As such, the RSBs become the upload bottleneck of the files which enlarge the download delay of vehicles. To address this effect requires to increase the upload capacity of RSBs.

VI. Discussions and Conclusion

In this paper, we proposed a distributed large-scale infrastructure for vehicular content distribution in urban areas.

The proposed infrastructure is formed by RSBs deployed across the city which are managed by individual entities at different locations and form an integrated system towards the global network utility. To enable RSBs to work in a fully distributed manner and accordingly make the proposed infrastructure scalable to any network size, we have introduced a random-walk based content replication scheme at RSBs. With extensive simulations, we have validated the performance of the proposed scheme.

The proposed infrastructure represents a new and practical solution on building the large-scale content distribution network for mobile users. Within this framework, there exist multiple interesting and open issues:

Connection to Internet: in this paper, we assume that RSBs are not connected to Internet. This can avoid the expensive bandwidth cost to RSB owners and encourage them to deploy RSBs. In practice, certain RSBs which are connected to the Internet, referred to as Internet-enabled RSBs, can be strategically deployed. In this case, by relying on vehicles to retrieve the Internet contents and transport the contents to different locations, limited Internet content distribution services can be provided through the RSB infrastructure. In this paradigm, the locations of these Internet-enabled RSBs are crucial to reduce the download delay of Internet contents to vehicles. For instance, the deployment of Internet-enabled RSBs should take the trajectories of vehicles and the density of the non Internet-enabled RSBs into consideration. From this perspective, [20] has shed important lights by investigating on the content replication issue, *i.e.*, to determine the optimal locations of contents stored in the deployed infrastructure based on the trajectories of vehicles.

Heterogenous Users: the proposed infrastructure can be extend to provide content distribution to different mobile users in different environments. For example, the RSBs can be deployed in a shopping mall to distribute store flyers to users with tablets, PDAs and laptops. In this case, different users may have different characteristics of mobility and requirements on the service quality. This dictates the network to take the distinct features of heterogenous user's QoS requirements into considerations.

Security Threat: without central control, the proposed infrastructure faces multiple security threats. For example, the buffer storage of RSBs can be abused to store and distribute harmful contents or virus to vehicles. Moreover, the contents stored in RSBs may also be polluted by garbage contents with misleading and mismatched titles. The content population is severe and has been extensively investigated in peer-to-peer networks [26], which however has not been addressed in the vehicular content distribution networks. To combat the security issues, it is necessary for RSBs to quickly identify and filter the harmful and spam contents [27].

APPENDIX

A. Proof of Theorem 1

According to Lemma 1, given the caching profile $\mathbf{B}$, we have that the usage of RSB buffer storage X satisfies (5). As $v = \sum_{i\in\mathbf{F}} x_i\kappa_i^2 \le \kappa E(X)$, where v and κ are as defined in Lemma 1. By substituting $v \le \kappa E(X)$ into (5), we have

$$P(X \ge E(X) + \psi) \le \exp\left(-\frac{\psi^2}{2(v+\kappa\psi/3)}\right) \le \exp\left(-\frac{\psi^2}{2(\kappa E(X)+\kappa\psi/3)}\right).$$

By assuming $L = E(X) + \psi$ and substituting it into (5), we have

$$\Pr(X \ge L) \le \exp\left(-\frac{(L-E(X))^2}{2(\kappa E(X) + \kappa(L-E(X))/3)}\right).$$

As such, the constraint of (4) can be achieved if

$$\exp\left(-\frac{(L-E(X))^2}{2(\kappa E(X) + \kappa(L-E(X))/3)}\right) \le \varepsilon. \quad \text{(A-1)}$$

By solving (A-1), we have that the constraint of (4) is satisfied if

$$E(X) = \sum_{i\in\mathbf{F}} x_i\kappa_i \le L - \kappa\frac{2}{3}\log\epsilon - \sqrt{\kappa^2\frac{4}{9}\log^2\epsilon - 2\kappa L\log\epsilon}.$$

B. Proof of Lemma 2

By applying the Taylor series expansion, the second order approximation of r as a function n can be represented as

$$r \approx r|_{\langle n\rangle} + (n-\langle n\rangle)\left.\frac{dr}{dn}\right|_{\langle n\rangle} + \frac{1}{2}(n-\langle n\rangle)^2\left.\frac{dr^2}{dn^2}\right|_{\langle n\rangle}. \quad \text{(B-1)}$$

By taking the expectation on both sides of (B-1) with respect to n, we have

$$\langle r\rangle \approx r|_{\langle n\rangle} + \frac{1}{2}\mathrm{Var}(n)\left.\frac{dr^2}{dn^2}\right|_{\langle n\rangle},$$

where $\mathrm{Var}(n)$ denotes the variance of n.

C. Proof of Lemma 3

Similar to the proof of Lemma 2, by applying the Taylor series expansion, the second order approximation of R as a function of n represented as

$$\begin{aligned} R \approx\ & b_i\left(G(\langle n\rangle) + (n-\langle n\rangle)\left.\frac{dG(n)}{dn}\right|_{\langle n\rangle}\right. \quad \text{(C-1)} \\ & \left. + \frac{1}{2}(n-\langle n\rangle)^2\left.\frac{dG^2(n)}{dn^2}\right|_{\langle n\rangle}\right) + (1-b_i)\, r, \end{aligned}$$

where $G(n) = \frac{C_{\mathsf{V2R}}}{n+1}$.

By taking the expectation of (C-1) on both sides with respect to n, we have

$$\begin{aligned} \langle R\rangle &\approx b_i\left(G(\langle n\rangle) + \frac{1}{2}\mathrm{Var}(n)\left.\frac{dG^2(n)}{dn^2}\right|_{\langle n\rangle}\right) + (1-b_i)\widetilde{\langle r\rangle} \\ &\approx b_i C_{\mathsf{V2R}}\left(\frac{1}{\langle n\rangle+1} + \frac{\mathrm{Var}(n)}{(\langle n\rangle+1)^3}\right) + (1-b_2)\widetilde{\langle r\rangle}. \end{aligned}$$

D. Proof of Theorem 2

The mean first passage time from state $(0,k)$ can be represented in a recursive manner as

$$\Gamma(0,k) = \frac{1}{\mu+\delta} + \frac{\delta}{\mu+\delta}\Gamma(0,k+1) + \frac{\mu}{\mu+\delta}\Gamma(1,k), \tag{D-1a}$$

$$\Gamma(1,k) = \frac{1}{\lambda+\gamma} + \frac{\gamma}{\lambda+\gamma}\Gamma(1,k+1) + \frac{\lambda}{\lambda+\gamma}\Gamma(0,k), \tag{D-1b}$$

for $0 \le k \le \kappa_i - 1$, and

$$\Gamma(0,\kappa_i) = \Gamma(1,\kappa_i) = 0. \tag{D-2}$$

In (D-1a), the first term on the right-hand-side represents the mean time that the tagged node spends in state $(0,k)$. With probability $\frac{\delta}{\mu+\delta}$, the tagged node transits to state $(0,k+1)$ which has the mean first passage time $\Gamma(0,k+1)$; with the rest probability, it transits to state $(1,k)$ which has the mean first passage time $\Gamma(1,k)$. (D-1b) is derived in the same manner.

As such, we have

$$\begin{aligned} &\lambda\Gamma(0,k) + \mu\Gamma(1,k) \qquad \text{(D-3)}\\ =\ & \frac{A}{B} + \frac{\lambda+\mu}{B}\left(\lambda\delta\Gamma(0,k+1) + \mu\gamma\Gamma(1,k+1)\right)\\ &+\frac{\delta\gamma}{B}\left(\lambda\Gamma(0,k+1) + \mu\Gamma(1,k+1)\right), \end{aligned}$$

where $A = (\lambda+\mu)^2 + \lambda\gamma + \mu\delta, B = \lambda\delta + \gamma\mu + \delta\gamma$.

In particular, via (D-1a) we have

$$\lambda\delta\Gamma(0,k) + \mu\gamma\Gamma(1,k) = (\kappa_i - k)(\lambda+\mu). \tag{D-4}$$

By substituting (D-4) to (D-3), we have

$$\begin{aligned} &\lambda\Gamma(0,k) + \mu\Gamma(1,k) \qquad \text{(D-5)}\\ =&\frac{A}{B} + \frac{(\kappa_i-k-1)(\lambda+\mu)^2}{\Pi}\\ &+\frac{\delta\gamma}{B}\left(\lambda\Delta(0,k+1) + \mu\Delta(1,k+1)\right)\\ =&\cdots\\ =&\sum_{i=0}^{\kappa_i-k-1}\left(\frac{\delta\gamma}{B}\right)^i\left(\frac{A}{B} + \frac{(\kappa_i-k-1)(\lambda+\mu)^2}{B}\right)\\ =&\frac{1-\left(\frac{\delta\gamma}{B}\right)^{\kappa_i-k}}{1-\frac{\delta\gamma}{B}}\left(\frac{A}{B} + \frac{(\kappa_i-k-1)(\lambda+\mu)^2}{B}\right). \end{aligned}$$

By plugging (D-5) into (12), we have

$$\begin{aligned} \tau_i &= \frac{1}{\lambda+\mu}\left(\lambda\Gamma(0,0) + \mu\Gamma(1,0)\right)\\ &= \frac{1}{\lambda+\mu}\frac{1-\left(\frac{\delta\gamma}{B}\right)^{\kappa_i}}{1-\frac{\delta\gamma}{B}}\left(\frac{A}{B} + \frac{(\kappa_i-1)(\lambda+\mu)^2}{B}\right). \end{aligned}$$

As $\frac{\delta\gamma}{B} < 1$, when κ_i is large, we have $\left(\frac{\delta\gamma}{B}\right)^{\kappa_i} \approx 0$, and accordingly,

$$\begin{aligned} \tau_i &= \frac{1}{\lambda+\mu}\frac{1}{1-\frac{\delta\gamma}{B}}\left(\frac{A}{B} + \frac{(\kappa_i-1)(\lambda+\mu)^2}{B}\right)\\ &= \frac{\gamma\lambda + \delta\mu + \kappa_i(\lambda+\mu)^2}{\gamma\mu(\lambda+\mu) + \lambda\delta(\lambda+\mu)} \end{aligned} \tag{D-6}$$

By substituting (16), $\widetilde{\langle R\rangle} = \gamma$ and $\widetilde{\langle r\rangle} = \delta$ into (D-6), we have

$$\tau_i = \frac{b_i(\Phi-\delta)\lambda + \delta(\lambda+\mu) + \kappa_i(\lambda+\mu)^2}{b_i(\Phi-\delta)\mu(\lambda+\mu) + \delta(\lambda+\mu)^2}.$$

E. Proof of Corollary 1

According to Theorem 2, we obtain the first and second order derivative of τ_i as

$$\begin{aligned} \frac{d\tau_i}{db_i} &= -\frac{(\Phi-\delta)\left[\kappa_i\mu(\lambda+\mu) + \delta(\mu-\lambda)\right]}{\left[b_i\mu(\Phi-\delta) + (\lambda+\mu)\right]^2}\\ \frac{d^2\tau_i}{db_i^2} &= \frac{2\mu(\Phi-\delta)^2\left[\kappa_i\mu(\lambda+\mu) + \delta(\mu-\lambda)\right]}{\left[b_i\mu(\Phi-\delta) + \delta(\lambda+\mu)\right]^3} \end{aligned}$$

The download delay τ_i is a convex function if $\frac{d^2\tau_i}{db_i^2} \ge 0$, *i.e.*, $\frac{\kappa_i}{\delta} \ge \frac{1}{\mu} - \frac{2}{\lambda+\mu}$.

F. Proof of Proposition 1

Evaluating the first and second derivatives of $U(\tau_i)$ on b_i, we have

$$\frac{dU(\tau_i)}{dx_i} = \frac{dU(\tau_i)}{d\tau_i}\frac{d\tau_i}{db_i} \tag{F-0a}$$

$$\frac{d^2U(\tau_i)}{db_i^2} = \frac{d^2U(\tau_i)}{d\tau_i^2}\left(\frac{d\tau_i}{db_i}\right)^2 + \frac{dU(\tau_i)}{d\tau_i}\frac{d^2\tau_i}{db_i^2} \tag{F-0b}$$

Since $\frac{dU(\tau_i)}{d\tau_i} \le 0$ and $\frac{d\tau_i}{db_i} \le 0$, with (F-1a) we have $\frac{dU(\tau_i)}{db_i} \ge 0$. Since $\frac{d^2U(\tau_i)}{d\tau_i^2} \le 0$, $\frac{dU(\tau_i)}{d\tau_i} \le 0$ and $\frac{d^2\tau_i}{db_i^2} \ge 0$, with (F-1b), we have $\frac{d^2U(\tau_i)}{db_i^2} \le 0$, and therefore, $U(\tau_i)$ is a concave function of b_i. As the network utility W, in general, is the weighted sum of the $U(\tau_i)$ over $i \in \mathbf{F}$, we have $\frac{d^2W}{db_i^2} = w_i\frac{d^2U(\tau_i)}{db_i^2} \le 0$. Therefore, W is also a concave function of b_i.

G. Optimal Solution of (7) According to KKT Conditions

As (7) is a convex optimization problem, the KKT conditions are both necessary and sufficient for the optimal solution. Let b_i^* denote the optimal solution of (7). Introducing Lagrangian multiplier ϖ for the constraint in (7), we list the KKT conditions of (7) as follows:

$$\left.\frac{\partial\mathcal{U}}{\partial b_i}\right|_{b_i^*} - \varpi\kappa_i = 0 \tag{G-1}$$

$$\varpi(\mathcal{L} - \kappa_i b_i^*) = 0 \tag{G-2}$$

$$\sum_{i\in\mathbf{F}}\kappa_i b_i^* \le \mathcal{L} \tag{G-3}$$

$$b_i^*, \gamma \ge 0,\ i \in \mathbf{F} \tag{G-4}$$

where $\mathcal{U} = -\sum_{i\in\mathbf{F}} p_i\tau_i$ as specified in (1) and (14)

Assume that $\varpi = 0$. Substituting it into (G-1), we have $p_i \left.\frac{\partial \tau_i}{\partial b_i}\right|_{b_i^*} = 0$. This is not feasible as $\frac{\partial \tau_i}{\partial b_i} < 0$ and $p_i > 0$ for all $i \in \mathbf{F}$. Therefore, from (G-2) we have $\gamma > 0$ and $\mathcal{L} - \sum_{i \in \mathbf{F}} b_i^* = 0$.

Substituting (13) into (G-1), we have

$$b_i^* = \frac{1}{\mu(\Phi-\delta)}\sqrt{\frac{p_i(\Phi-\delta)\left[\kappa_i\mu(\lambda+\mu)+\delta(\mu-\lambda)\right]}{\varpi\kappa_i}} - \frac{\delta(\lambda+\mu)}{\mu(\Phi-\delta)}. \quad \text{(G-5)}$$

Together with $\mathcal{L} - \sum_{i \in \mathbf{F}} b_i^* = 0$, we have

$$b_i^* = \mathcal{L}\frac{\sqrt{p_i\left[\kappa_i\mu(\lambda+\mu)+\delta(\mu-\lambda)\right]}}{\sum_{j\in\mathbf{F}}\kappa_j\sqrt{p_j\left[\kappa_j\mu(\lambda+\mu)+\delta(\mu-\lambda)\right]}} + \left(\frac{\sqrt{p_i\left[\kappa_i\mu(\lambda+\mu)+\delta(\mu-\lambda)\right]}\sum_{i\in\mathbf{F}}\kappa_i}{\sum_{j\in\mathbf{F}}\kappa_j\sqrt{p_j\left[\kappa_j\mu(\lambda+\mu)+\delta(\mu-\lambda)\right]}} - 1\right) \cdot \frac{\delta(\lambda+\mu)}{\mu(\Phi-\delta)}, \quad \text{(G-6)}$$

and

$$\varpi = \left(\frac{\sum_{i\in\mathbf{F}}\sqrt{\kappa_i p_i\left[\kappa_i\mu(\lambda+\mu)+\delta(\mu-\lambda)\right]}}{\mathcal{L}\mu+\delta(\lambda+\mu)\sum_{i\in\mathbf{F}}\kappa_i}\right)^2$$

H. Transition Probability in the Random Walk Algorithm

The target of the random walk algorithm is to select a file i from the file graph with the probability b_i^{d} shown in (15).

Assume that there are totally V nodes presenting in the network and a_i of them having file i stored. Therefore, there are totally $a_i|V|$ copies of file i in the file graph. To select file i with probability b_i^{d}, one should sample each copy of file i in the graph with probability

$$\pi_i = \frac{b_i^{\mathsf{d}}}{a_i V}. \quad \text{(H-1)}$$

Using the Metropolis-Hasting algorithm, the transition of random walk constitutes two steps. In the first step, a candidate file, *e.g.*, m, is selected from the neighboring files of the current file, *e.g.*, n, which holds the walker based on the proposal probability

$$\alpha_{mn} = \frac{1}{s_m+1}, \quad \text{(H-2)}$$

where s_m denotes the fanout of file m in the file graph. A neighboring file of file n is the file which is connected to file n in the file graph.

In the second step, file m is accepted as the next hop of the walker with the acceptance probability as

$$q_{mn} = \min\left\{\frac{\pi_n\alpha_{nm}}{\pi_m\alpha_{mn}}, 1\right\} = \min\left\{\frac{a_m b_n^{\mathsf{d}}(s_m+1)}{a_n b_m^{\mathsf{d}}(s_n+1)}, 1\right\},$$

with the rest probability the walk will sojourn in file n for one hop.

Therefore, the transition probability from file m to file n is

$$P_{mn} = \alpha_{mn}q_{mn} = \begin{cases} \frac{1}{s_m+1}\min\left\{\frac{a_m b_n^{\mathsf{d}}(s_m+1)}{a_n b_m^{\mathsf{d}}(s_n+1)}, 1\right\}, & m \neq n, \\ 1 - \sum_{m\neq n} P_{mn}, & m = n. \end{cases} \quad \text{(H-3)}$$

REFERENCES

[1] H. T. Cheng, H. Shan, and W. Zhuang, "Infotainment and Road Safety Service Support in Vehicular Networking: From a Communication Perspective," *Mechanical Systems and Signal Processing, Special Issue on Integrated Vehicle Dynamics*, vol. 25, no. 6, pp. 2020 – 2038, August 2011.

[2] N. Post, "Toronto Hydro Assailed for City-Wide WiFi Plan," Canada.com, March 7 2006.

[3] M. Reardon, "Cities Deploying Wi-Fi Face Challenges," CNET News, May 1 2006.

[4] D. Hadaller, S. Keshav, T. Brecht, and S. Agarwal, "Vehicular Opportunistic Communication Under the Microscope," in *Proc. of ACM MobiSys*, 2007.

[5] B. Yu and F. Bai, "ETP: Encounter Transfer Protocol for Opportunistic Vehicle Communication," in *Proc. of IEEE Infocom*, 2011.

[6] F. Bai and B. Krishnamachari, "Spatio-Temporal Variations of Vehicle Traffic in VANETs: Facts and Implications," in *Proc. of ACM VANETs*, 2009.

[7] M. Asefi, J. W. Mark, and X. Shen, "A Mobility-Aware and Quality-Driven Retransmission Limit Adaptation Scheme for Video Streaming over VANETs," *IEEE Transactions on Wireless Communications*, vol. 11, no. 5, pp. 1817 – 1827, May 2012.

[8] A. Nandan, S. Das, G. Pau, M. Gerla, and M. Y. Sanadidi, "Co-operative Downloading in Vehicular Ad-Hoc Wireless Networks," in *Proc. of IEEE/IFIP WONS*, 2005.

[9] U. Lee, J. S. Park, J. Yeh, G. Pau, and M. Gerla, "Code Torrent: Content Distribution Using Network Coding in VANET," in *Proc. of ACM Mobishare*, 2006.

[10] H. Liang and W. Zhuang, "Cooperative Data Dissemination via Roadside WLANs," *IEEE Communications Magazine*, vol. 50, no. 4, pp. 68 – 74, May 2012.

[11] Y. Zhang, J. Zhao, and G. Cao, "On Scheduling Vehicle-Roadside Data Access," in *Proc. of ACM VANET*, 2007.

[12] O. Trullols-Cruces, M. Fiore, and J. Barcelo-Ordinas, "Cooperative Download in Vehicular Environments," *IEEE Transactions on Mobile Computing*, no. 99, pp. 663 – 678, April 2011.

[13] S. H. Lee, U. Lee, K. W. Lee, and M. Gerla, "Content Distribution in VANETs using Network Coding: the Effect of Disk I/O and Processing O/H," in *Proc. of IEEE SECON*, 2008.

[14] M. Li, Z. Yang, and W. Lou, "Codeon: Cooperative Popular Content Distribution for Vehicular Networks using Symbol Level Network Coding," vol. 29, no. 1, pp. 223 – 235, January 2010.

[15] Q. Yan, M. Li, Z. Yang, W. Lou, and H. Zhai, "Throughput Analysis of Cooperative Mobile Content Distribution in Vehicular Network using Symbol Level Network Coding," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 2, pp. 484 – 492, February 2012.

[16] F. Ye, S. Roy, and H. Wang, "Efficient Data Dissemination in Vehicular Ad Hoc Networks," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 4, pp. 769 – 779, May 2012.

[17] Y. Zhang and G. Cao, "V-PADA: Vehicle-Platoon Aware Data Access in VANETs," *IEEE Transactions on Vehicular Technology*, no. 99, pp. 2326 – 2339, June 2011.

[18] U. G. Acer, P. Giaccone, D. Hay, G. Neglia, and S. Tarapiah, "Timely Data Delivery in a Realistic Bus Network," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 3, pp. 1251 – 1265, March 2012.

[19] A. Nandan, S. Tewari, S. Das, M. Gerla, and L. Kleinrock, "AdTorrent: Delivering Location Cognizant Advertisements to Car Networks," in *Proc. of IEEE/IFIP WONS*, 2006.

[20] Y. Huang, Y. Gao, K. Nahrstedt, and W. He, "Optimizing File Retrieval in Delay-Tolerant Content Distribution Community," in *Proc. of IEEE ICDCS*, 2009.

[21] Y. Zhuang, J. Pan, Y. Luo, and L. Cai, "Time and Location-Critical Emergency Message Dissemination for Vehicular Ad-Hoc Networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 1, pp. 187 – 196, January 2011.

[22] F. R. K. Chung and L. Lu, *Complex Graphs and Networks*. Amer Mathematical Society, 2006.

[23] U. Lee, E. Magistretti, M. Gerla, P. Bellavista, and A. Corradi, "Dissemination and Harvesting of Urban Data Using Vehicular Sensing Platforms," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 2, pp. 882 – 901, February 2009.

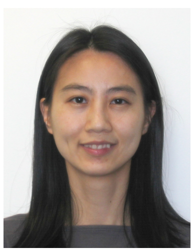
[24] "U.S. Census Bureau TIGER/Line Shapefiles & Files," Retrieved in February 2012. [Online]. Available: http://www.census.gov/geo/www/tiger/

[25] J. Haerri, F. Filali, C. Bonnet, and M. Fiore, "VanetMobiSim: Generating Realistic Mobility Patterns for VANETs," in *Proc. of ACM VANET*, 2006.

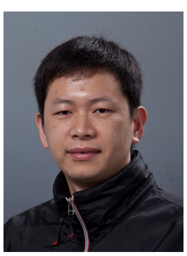
[26] J. Liang, R. Kumar, Y. Xi, and K. W. Ross, "Pollution in P2P File Sharing Systems," in *Proc. of IEEE Infocom*, 2005.

[27] R. Lu, X. Lin, T. H. Luan, X. Liang, X. Li, L. Chen, and X. Shen, "PReFilter: An Efficient Privacy-preserving Relay Filtering Scheme for Delay Tolerant Networks," in *Proc. of IEEE Infocom*, 2012.

Tom H. Luan received the B.E. degree in Xi'an Jiaotong University, China in 2004, the M.Phil. degree in the Hong Kong University of Science and Technology, Kowloon, Hong Kong in 2007, and the Ph.D. degree at the University of Waterloo, ON, Canada in 2012. His current research interests focus on wired and wireless multimedia streaming, QoS routing in multihop wireless networks, peer-to-peer streaming and vehicular network design.



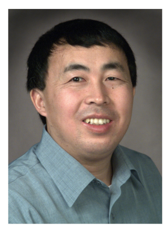
Lin X. Cai received the M.A.Sc. and PhD degrees in electrical and computer engineering from the University of Waterloo, Ontario, Canada, in 2005 and 2010, respectively. She was working as a postdoctoral research fellow in electrical engineering department at Princeton University in 2011. Currently, she is a senior engineer in US wireless R&D center in Huawei Technologies Inc. Her research interests include green communication and networking, resource management for broadband multimedia networks, and cross-layer optimization and QoS provisioning.



Jiming Chen (IEEE M'08-SM'11) received B.Sc degree and Ph.D degree both in Control Science and Engineering from Zhejiang University in 2000 and 2005, respectively. He was a visiting researcher at INRIA in 2006, National University of Singapore in 2007, and University of Waterloo from 2008 to 2010. Currently, he is a full professor with Department of control science and engineering, and the coordinator of group of Networked Sensing and Control in the State Key laboratory of Industrial Control Technology, Vice Director of Institute of Industrial Process Control at Zhejiang University, China. He currently serves associate editors for several international Journals including IEEE Transactions on Parallel and Distributed System, IEEE Transactions on Industrial Electronics, IEEE Network, IET Communications, *etc.* He was a guest editor of IEEE Transactions on Automatic Control, Computer Communication (Elsevier), Wireless Communication and Mobile Computer (Wiley) and Journal of Network and Computer Applications (Elsevier). He also served/serves as Ad hoc and Sensor Network Symposium Co-chair, IEEE Globecom 2011; general symposia Co-Chair of ACM IWCMC 2009 and ACM IWCMC 2010, WiCON 2010 MAC track Co-Chair, IEEE MASS 2011 Publicity Co-Chair, IEEE DCOSS 2011 Publicity Co-Chair, IEEE ICDCS 2012 Publicity Co-Chair, IEEE ICCC 2012 Communications QoS and Reliability Symposium Co-Chair, IEEE SmartGridComm The Whole Picture Symposium Co-Chair, IEEE MASS 2013 Local Chair, Wireless Networking and Applications Symposium Co-chair, IEEE ICCC 2013 and TPC member for IEEE ICDCS'10,'12,'13, IEEE MASS'10,11,'13, IEEE SECON'11,'12 IEEE INFOCOM'11,'12,'13, *etc.*

Fan Bai (General Motors Global R&D) is a Senior Researcher in the Electrical & Control Integration Lab., Research & Development and Planning, General Motors Corporation, since Sep., 2005. Before joining General Motors research lab, he received the B.S. degree in automation engineering from Tsinghua University, Beijing, China, in 1999, and the M.S.E.E. and Ph.D. degrees in electrical engineering, from University of Southern California, Los Angeles, in 2005.

His current research is focused on the discovery of fundamental principles and the analysis and design of protocols/systems for next-generation Vehicular Ad hoc Networks (VANET), for safety, telematics and infotainment applications. Dr. Bai has published about 40 book chapters, conference and journal papers, including Mobicom, INFOCOM, MobiHoc, SECON, ICC, Globecom, WCNC, JSAC, IEEE Wireless Communication Magazine, IEEE Communication Magazine and Elsevier AdHoc Networks Journal. In 2006, he received Charles L. McCuen Special Achievement Award from General Motors Corporation in recognition of extraordinary accomplishment in area of vehicle-to-vehicle communications for drive assistance & safety. He serves as Technical Program Co-Chairs for IEEE WiVec 2007 and IEEE MoVeNet 2008. He is an associate editor of IEEE Transaction on Vehicular Technology and IEEE Transaction on Mobile Computing, and serves as guest editors for IEEE Wireless Communication Magazine, IEEE Vehicular Technology Magazine and Elsevier AdHoc Networks Journal. He is also serving as a Ph.D. supervisory committee member at Carnegie Mellon University and University of Illinois-Urban Champaign.



Xuemin (Sherman) Shen (IEEE M'97-SM'02-F09) received the B.Sc.(1982) degree from Dalian Maritime University (China) and the M.Sc. (1987) and Ph.D. degrees (1990) from Rutgers University, New Jersey (USA), all in electrical engineering.

He is a Professor and University Research Chair, Department of Electrical and Computer Engineering, University of Waterloo, Canada. He was the Associate Chair for Graduate Studies from 2004 to 2008. Dr. Shen's research focuses on resource management in interconnected wireless/wired networks, wireless network security, wireless body area networks, vehicular ad hoc and sensor networks. He is a co-author/editor of six books, and has published many papers and book chapters in wireless communications and networks, control and filtering. Dr. Shen served as the Technical Program Committee Chair for IEEE VTC'10 Fall, the Symposia Chair for IEEE ICC'10, the Tutorial Chair for IEEE VTC'11 Spring and IEEE ICC'08, the Technical Program Committee Chair for IEEE Globecom'07, the General Co-Chair for Chinacom'07 and QShine'06, the Chair for IEEE Communications Society Technical Committee on Wireless Communications, and P2P Communications and Networking. He also serves/served as the Editor-in-Chief for IEEE Network, Peer-to-Peer Networking and Application, and IET Communications; a Founding Area Editor for IEEE Transactions on Wireless Communications; an Associate Editor for IEEE Transactions on Vehicular Technology, Computer Networks, and ACM/Wireless Networks; and the Guest Editor for IEEE JSAC, IEEE Wireless Communications, IEEE Communications Magazine, and ACM Mobile Networks and Applications.

Dr. Shen is a registered Professional Engineer of Ontario, Canada, an IEEE Fellow, a Fellow of the Canadian Academy of Engineering, a Fellow of Engineering Institute of Canada, and a Distinguished Lecturer of IEEE Vehicular Technology Society and Communications Society.