Low-Latency and Fresh Content Provision in Information-Centric Vehicular Networks

Shan Zhang, *Member, IEEE*, Junjie Li, Hongbin Luo, *Member, IEEE*, Jie Gao, *Member, IEEE*, Lian Zhao, *Senior Member, IEEE*, and Xuemin (Sherman) Shen, *Fellow, IEEE*

Abstract—In this paper, the content service provision of information-centric vehicular networks (ICVNs) is investigated from the aspect of mobile edge caching, considering the dynamic driving-related context information. To provide up-to-date information with low latency, two schemes are designed for cache update and content delivery at the roadside units (RSUs). The roadside unit centric (RSUC) scheme decouples cache update and content delivery through bandwidth splitting, where the cached content items are updated regularly in a round-robin manner. The request adaptive (ReA) scheme updates the cached content items upon user requests with certain probabilities. The performance of both proposed schemes are analyzed, whereby the average age of information (AoI) and service latency are derived in closed forms. Surprisingly, the AoI-latency trade-off does not always exist, and frequent cache update can degrade both performances. Thus, the RSUC and ReA schemes are further optimized to balance the AoI and latency. Extensive simulations are conducted on SUMO and OMNeT++ simulators, and the results show that the proposed schemes can reduce service latency by up to 80 percent while guaranteeing content freshness in heavily loaded ICVNs.

Index Terms—Mobile edge caching, age of information (AoI), latency, vehicular networking

1 Introduction

Vehicular communications feature high mobility, short-lived link connectivity, and rapidly changing network topology, posing great challenges to IP-based solutions in terms of efficiency and reliability [1]. In this context, Information Centric Vehicular Networking (ICVN) is emerging as a promising paradigm, which supports the receiver-driven content retrieval without requiring the conventional sender-driven end-to-end connectivity [2]. Furthermore, ICVN can utilize the storage resources of roadside units (RSUs) or vehicles to realize mobile edge caching, bringing benefits including reduced end-to-end latency, high transmission efficiency, and enhanced system reliability [3]. However, in practice, mobile edge caching usually faces the contradiction of limited storage resource and massive data, raising fundamental issues including where to cache, what to cache, and how to update [4], [5], [6], [7]. Specifically, cache update is critical in the dynamic ICVN environment. On the one hand, the popularity of content items may change with time, since new items keep generating (such as news and entertainment information) [8], [9]. On the other hand, there exists extensive driving-related context information which may vary with time, such as the position of surrounding vehicles, notification of accidents, status of traffic lights, availability of parking lot, and promotion of stores [10]. For the former case, the newly generated popular content items should be cached and replace the unpopular ones, to keep high hit rate at the network edge. The latter case requires to download new versions of the cached items from time to time to guarantee the content effectiveness. Extensive efforts have been devoted to cache update considering the content popularity variation, including popularity prediction and content replacement algorithm design [11], [12], [13]. However, the dynamic content variations of cached items has been seldom considered.

In this work, we investigate the joint cache updating and content delivery scheduling scheme of RSUs in ICVNs, considering the dynamic context information services. The objective is to provide fresh information to vehicles rapidly on demand. A typical ICVN scenario is considered, which consists of producers generating content items, cache-enabled RSUs providing content services, and moving vehicles randomly raising content requests. As the context information may change with the driving environment, the producers continuously generate new versions of content items to reflect the real-time status. Meanwhile, RSUs update the cache timely to deliver fresh content to vehicles. Notice that cache update will consume additional bandwidth resources, which may degrade the content delivery efficiency and introduce longer service latency. Thus, cache update and content delivery should be jointly scheduled to balance content freshness and service latency. In this regard, we propose two schemes, i.e., the RoadSide Unit Centric (RSUC) scheme and the Request Adaptive (ReA) scheme, based on the named data

- *Shan Zhang, Junjie Li, and Hongbin Luo are with the State Key Laboratory of Software Development Environment, School of Computer Science and Engineering, Beihang University, Beijing 100191, China. E-mail: {zhangshan18, ljj0618, luohb}@buaa.edu.cn.*
- *Jie Gao is with the Department of Electrical and Computer Engineering, Marquette University, Milwaukee, WI 53233 USA. E-mail: j.gao@marquette.edu.*
- *Lian Zhao is with the Department of Electrical, Computer, and Biomedical Engineering, Ryerson University, Toronto M5B 2K3, ON, Canada. E-mail: l5zhao@ryerson.ca.*
- *Xuemin (Sherman) Shen is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo N2L 3G1, ON, Canada. E-mail: sshen@uwaterloo.ca.*

Manuscript received 25 Nov. 2019; revised 1 Aug. 2020; accepted 14 Sept. 2020. Date of publication 18 Sept. 2020; date of current version 4 Apr. 2022. (Corresponding author: Hongbin Luo.) Digital Object Identifier no. 10.1109/TMC.2020.3025201

networking (NDN) architecture. The RSUC scheme decouples content update and content delivery by splitting the bandwidth. All cached items are updated in a round-robin manner, while the RSU will serve the requests with the current version of cached content items in a First-Come-First-Serve (FCFS) policy. The ReA scheme updates cached items along with content delivery. Specifically, a request triggers cache update with a certain probability, and then the RSU will fetch the new version of the request item before delivery. Two substantially different performance metrics, i.e., Age of Information (AoI) and service latency are adopted to analyze the proposed schemes. AoI, defined as the time elapsed since the generation of the content, depicts the freshness of the content. Instead, service latency characterizes the response speed of the RSU, i.e., the time needed for a vehicle to receive the content after raising a request. By applying stochastic process and queueing theory, the average AoI and service latency are derived in closed forms under both schemes. Analytical results show that the content freshness and service latency do not always show a trade-off, and the frequent cache update may increase the average AoI and latency simultaneously. Therefore, we further optimize the RSUC and ReA schemes to balance the AoI and latency. Extensive simulations have been conducted using the Monte Carlo method on the OMNeT++ platform to validate the theoretical analysis and evaluate the performance of the two schemes under different scenarios and system parameters. In addition, real-trace system level simulations are conducted using the city map of Erlangen, Germany, based on SUMO and OMNeT++ simulation platforms. Simulation results show that the ReA and RSUC schemes can reduce latency by 65 and 80 percent compared with the conventional pull-based NDN method, and the RSUC scheme can improve the throughput by around one fold compared with the push-based data forwarding method under the given AoI and latency requirements [14].

The main contributions of this work are as follows:

- The joint design of cache update and content delivery is studied under the ICVN architecture considering the content dynamics, where content freshness and service latency are both guaranteed to enhance the quality of experience;
- The average AoI and latency are derived under the proposed schemes, whereby the AoI-latency interplay is quantified through theoretical analysis;
- The proposed schemes are further optimized for AoI-latency balance on the demand of applications.

The remaining of this paper is organized as follows. Section 2 reviews related works. Section 3 introduces the architecture of ICVN and proposes cache update and content delivery schemes. The performances of the proposed schemes are analyzed in Section 4, whereby the interplay between AoI and latency performances is studied in Section 5. Simulation results are provided in Section 6, followed by conclusions in Section 7.

2 Literature Review

This section provides a review on the design of ICVN architecture, in-network caching, AoI and timely content services to highlight the novelty of this work.

2.1 Vehicular Information Centric Network Architecture

Extensive efforts have been devoted to the ICVN architecture design. Classical Information Centric Networking (ICN) architectures, such as Content Centric Networking (CCN), Named Data Networking (NDN), and MobilityFirst, have been enhanced to adapt to vehicular communication features like high mobility and wireless link uncertainty [15], [16], [17]. Grewe *et al.* have shown that the CCN/NDN based approaches can satisfy most of the vehicular communication requirements like naming, data dissemination, safety and security [2]. Furthermore, Grassi *et al.* have built an NDN-based vehicular communication prototype [18]. Content dissemination is a critical management issue due to the vehicular mobility and link unreliability [19]. A location-based content forwarding scheme has been proposed to allow vehicles fetch data from multiple potential carriers for reliability, which is more robust to vehicle topology variations and link disruptions [20]. Boukerche *et al.* have suggested to use neighboring vehicles with higher link reliability, to improve content delivery rate and reduce duplicated network transmissions [21]. According to the conventional NDN architecture, content items are delivered in a pull-based manner, and the producer can only send the content after receiving a user request. However, such pull-based method may cause long content retrieval latency, and cannot meet the requirements of time-critical applications. Accordingly, some recent studies have also worked on push-based NDN architectures, which enables producers to inject time-critical content to RSUs or even vehicles prior to requests [14], [22].

2.2 In-network Caching in Vehicular Networks

In-network caching is one of the key design issues for ICN architectures, and has been extensively studied in the static Internet environments [23]. However, this problem needs to be revisited for the ICVN considering the short-lived connections, time-varying topology, dynamic context information and applications [24]. The very recent works have started exploration in this area, including RSU caching [25], [26] vehicle caching [8], [27], [28], [29].

Caching at RSUs can address the backhaul congestion issues, which is beneficial for both network deployment and service quality enhancement [25]. Efficient RSU content placement algorithm has been proposed to maximize the content retrieval probability by solving an integer linear programming problem, with the cache size optimized [26]. The results have shown that RSU caching is more effective compared with the core-network caching, especially in case of low RSU density or high vehicle density. In addition to RSU caching, the rich storage resources of vehicles can be also utilized. Location-based content items can be pushed to the cache of vehicles prior to requests based on the driving trajectory, which can better support vehicle mobility [27]. Cache-enabled vehicles can also form groups for efficient content delivery in a crowdsourcing manner [28]. Specifically, it is more efficient to cache contents on the vehicles which may stay in hot regions for longer time or have more social connectivities [29]. The very few works have studied cache update based on the dynamics of content popularity

TABLE 1
Typical on-Road Context Infomration and Service Requirements

Catergory	Content		Requirements
Driving-related	Surrounding vehicles	Position, mobility (speed, steering, headway), condition (gas, engine, lights, wheel pressure)	Latency ≤ 100 ms AoI ≤ 100 ms
	Road condition	Map, signs, signals, constructions, restrictions	
	Emergency	Overtaking, lane changing, merge, brake alarm, accident warning, pedestrians	
	Macro scale	Traffic density, traffic jam, real-time map, parking lot, ETC	Latency ≤ 500 ms AoI ≤ 1s
Travel experience	On-road social networks, air and weather conditions, restaurants and promotions		Real-time/elastic

[8], whereas the dynamic variation of content information has been merely considered.

2.3 Age of Information and Timely Service

The concept of AoI was first introduced to capture the requirement of time-critical vehicular safety applications [30]. In specific, AoI is defined as the time elapsed since the generation of the corresponding content, which quantifies the freshness of knowledge we have about a remote time-varying source [31]. Recent works have adopted this new concept to evaluate the performance of dynamic systems, and accordingly new system scheduling and management methods have been proposed. The AoI performance has been analyzed theoretically by applying queueing models in case of single source and multiple sources under different service mode [32], [33], [34]. Kam *et al.* have further utilized multi-server queueing model to reflect service priority and multi-path transmission diversity [35]. AoI-optimal content update schemes have been proposed for single-source and multi-source heterogeneous systems [36], [37]. Furthermore, AoI has been adopted for the metric of cloud game applications to achieve timely video frame transmissions [38].

The very recent works have implemented the AoI in mobile edge caching [39], [40], [41]. Considering that the popularity of a content item may fade with time, the authors have built a model of request rate based on both historical request rate and content AoI, whereby the optimal content update policy has been devised [39]. However, the dynamic variation of content is ignored. [40] and [41] have considered the content dynamics in mobile edge caching, which are the most related to this work. Yates *et al.* have studied the content update problem for a local cache system in [40] to minimized the average AoI of cached content items. An optimization problem is formulated and solved in an approximated way with relaxation. The main differences between this work and [40] are three-fold: (1) we aim at minimizing the AoI of user-received content instead of local-cached content, (2) the content caching and delivery are jointly optimized, and (3) the service latency is also considered in addition to AoI to provide better quality of experience. A cache-assisted lazy update and delivery (CALUD) scheme has been proposed in our preliminary work [41], which was the first to study the interplay between latency and content freshness performances from the edge caching aspect. Nevertheless, the CALUD scheme is limited to the ideal wireless channel assumption, and each source utilizes dedicated bandwidth in a low-efficient way.

3 Information-Centric Vehicular Network

We focus on the on-road context information, which can be further classified based on the applications and QoS requirements, as listed in Table 1 [10]. The driving-related context information helps to make driving decisions based on the surrounding and macro-scale environment, while the travel-related context information can enrich the travel experience of passengers. The considered ICVN scenario is illustrated in Fig. 1. The context information is collected by distributed sensors or information acquisition systems (called *producers*), and then published to the RSU for future use. Denote by $\mathcal{S} = \{1, 2, \ldots, S\}$ the set of content items generated by all producers, where $S = |\mathcal{S}|$ and each content item has the same size L. Vehicles are content *consumers* which raise requests randomly. The request arrival process of the content item s is modeled as a Poisson process of rate λ_s. A cache-enabled RSU is deployed, and works as a intermediate *content router* to forward the requested content items from producers to vehicle consumers on demand. We propose RSUC and ReA schemes based on the NDN architecture, and the service process of different schemes are illustrated in Fig. 2.

3.1 Conventional NDN Scheme

The conventional NDN architecture adopts a pull-based communication scheme, wherein a source node can only send the content after receiving a request. Accordingly, the producer only publish the name of generated content items to the RSU. Although NDN provides in-network caching opportunities, the dynamics of content items is not considered, and cannot be directly applied to the context information. In this regard, for each request, we consider that the

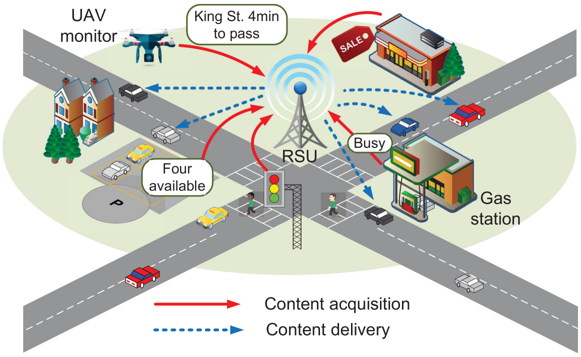


Fig. 1. The considered information-centric vehicular networking scenario.

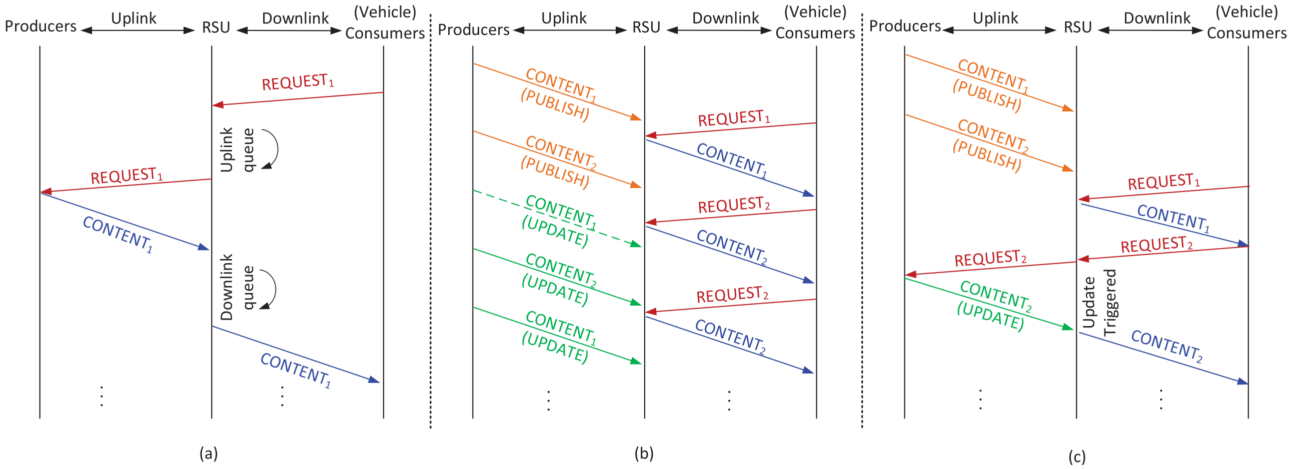


Fig. 2. Content service process: (a) conventional pull-based NDN, (b) proposed RSUC scheme, and (c) proposed ReA scheme.

RSU will always pull the up-to-date content versions from the producer upon requests to avoid providing outdated information, as shown in Fig. 2a. Denote by B the available bandwidth of the RSU, which is divided into uplink and downlink channels orthogonally with a splitting ratio β. The uplink connects the RSU with producers for content fetching, while the downlink connects the RSU and vehicle users for content delivery. Then, the service process of the RSU can be modeled as a tandem queueing system with two servers. The uplink and downlink average transmission rates are given by[1]

$$\begin{aligned}\mu_{\mathrm{UL}} &= \frac{\beta B}{L}\log_2(1+\gamma_{\mathrm{UL}}) \triangleq \beta R_{\mathrm{UL}},\\ \mu_{\mathrm{DL}} &= \frac{(1-\beta)B}{L}\log_2(1+\gamma_{\mathrm{DL}}) \triangleq (1-\beta) R_{\mathrm{DL}},\end{aligned} \tag{1}$$

respectively, where γ_{UL} and γ_{DL} are the average received signal to interference and noise ratio (SINR) of the uplink and downlink, R_{UL} and R_{DL} are the normalized uplink and downlink service rates with full bandwidth allocation.

3.2 RSU-Centric Scheme

The RSUC scheme decouples cache content fetching and delivery. Unlike the conventional NDN method, the RSUC scheme is push-based, where the producers can send the content items and new versions to the RSU prior to user requests, as shown in Fig. 2b. In the initialization stage, the producer will broadcast a message involving the information of generated content item. The RSU will send a request when receiving the message, and the produce will push the content to the RSU for cache. Then, the RSU use the uplink for cache update, wherein the producers push the up-to-date content versions to the RSU for future use in a round-robin manner. Meanwhile, the downlink is used for content delivery, wherein the RSU will directly send the cached content version to vehicles through one-hop transmission on demand. The uplink and downlink work independently, using orthogonal bandwidth βB and $(1-\beta)B$, respectively.

The RSUC scheme can guarantee the average AoI requirement by adjusting the bandwidth splitting ratio β. Consider a content item cached in the RSU. The AoI varies with the uplink cache update dynamically, as shown in Fig. 2. Assume the producers can generate new content versions on demand, i.e., zero AoI at the producer. Accordingly, the AoI of a cached item will be set to the uplink transmission time once updated. Then, the AoI increases with time linearly until the next update, according to the definition of AoI [30]. Intuitively, the average AoI can be reduced with more frequent cache update, requiring high bandwidth splitting ratio β. However, the available resource for content delivery will be reduced, degrading the service latency. Therefore, the bandwidth splitting ratio β is the key to tune the trade-off between AoI and latency, which will be analyzed in the next section.

3.3 Request-Adaptive Scheme

Although the RSUC scheme can realize one-hop content retrieval, the cache update cost may be high. Specifically, some pro-actively pushed content versions may not be used, as shown by the dashed line in Fig. 2b. Thus, we further introduce the ReA scheme for efficient cache update, by exploiting the request arrival information. Under the ReA scheme, the generated content items are pushed to the RSU at the beginning, whereas the cache update is pull-based and depends on vehicle requests, as shown in Fig. 2c. When a vehicle raises a request of item s, s will be updated with probability p_s at the RSU cache. If the update is not triggered, the RSU will directly send the cached content to the vehicle. Otherwise, the RSU pull the new version from the producer and then delivers the content. Under the ReA scheme, the update frequency of item s depends on both update probability p_s and request arrival rate λ_s, given by $p_s\lambda_s$. Thus, the average AoI requirement of item s can be guaranteed by adjusting p_s. Compared with the RSUC scheme, the content retrieval latency will be longer when the cache update is triggered under the ReA scheme, due to the two-hop transmission. However, the ReA scheme can conduct finer-grained update control based on content popularity.

4 Freshness and Latency Analysis

This section analyzes the performances of three schemes.

[1] The transmission rates can be modified based on the adopted physical layer technologies, while the analysis method still applies.

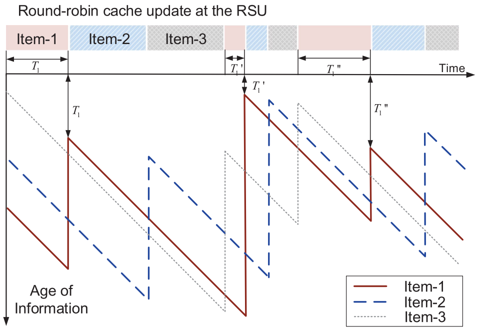


Fig. 3. Illustration of AoI variation for the RSU cached content items under the RSUC scheme.

4.1 Conventional Scheme Analysis

The service model of the conventional NDN pull-based scheme is shown in Fig. 4. The service time of both servers are considered to follow independent exponential distributions to reflect the randomness of wireless transmission. Accordingly, the queueing system of Fig. 4 is a Jackson queueing network, and the equilibrium joint probability distribution of queue lengths has a product-form solution. Thus, the uplink and downlink transmissions can be analyzed independently as two independent M/M/1 queues with equivalent arrival rate $\Lambda = \sum_{s=1}^{S} \lambda_s$. Therefore, the average latency to serve a content request is given by

$$\begin{aligned}\bar{D}_{\text{con}} &= \frac{1}{\mu_{\text{UL}} - \Lambda} + \frac{1}{\mu_{\text{DL}} - \Lambda} \\ &= \frac{1}{\beta R_{\text{UL}} - \Lambda} + \frac{1}{(1-\beta) R_{\text{DL}} - \Lambda},\end{aligned} \tag{2}$$

where the two parts correspond to cache update and content delivery, respectively. The average latency $\bar{D}_{\text{con}}$ increases with the total traffic arrival rate Λ and decreases with the normalized uplink and downlink transmission rates. Furthermore, the bandwidth splitting ratio β also influences the latency, and can be optimized by solving the following problem:

$$(\text{P1}) \quad \min_{\beta} \quad \frac{1}{\beta R_{\text{UL}} - \Lambda} + \frac{1}{(1-\beta) R_{\text{DL}} - \Lambda}, \tag{3a}$$

$$s.t. \quad \beta R_{\text{UL}} \geq \Lambda, \tag{3b}$$

$$(1-\beta) R_{\text{DL}} \geq \Lambda, \tag{3c}$$

$$0 \leq \beta \leq 1, \tag{3d}$$

where the objective is to minimize the average latency, constraints (3 b) and (3 c) guarantee the stability of the uplink and downlink queues, respectively. By analyzing problem (P1), we obtain two important performance metrics: (1) the network capacity, i.e., the maximal request arrival rate that can be accommodated, and (2) the minimal average service latency with the optimal bandwidth splitting, given as Theorem 1.

Theorem 1. *Under the conventional pull-based NDN scheme, the network capacity $\hat{\Lambda}_{\text{con}}$ is given by*

$$\hat{\Lambda}_{\text{con}} = \frac{1}{\frac{1}{R_{\text{UL}}} + \frac{1}{R_{\text{DL}}}}, \tag{4}$$

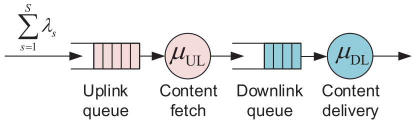


Fig. 4. Service model of the conventional NDN method.

and the minimal average service latency for the given traffic arrival rate Λ is

$$\bar{D}^*_{\text{con}} = \frac{\left(\frac{1}{\sqrt{R_{\text{UL}}}} + \frac{1}{\sqrt{R_{\text{DL}}}}\right)^2}{1 - \Lambda\left(\frac{1}{R_{\text{UL}}} + \frac{1}{R_{\text{DL}}}\right)}. \tag{5}$$

Proof. Constraints (3 b) and (3 c) can be written a

$$\frac{\Lambda}{R_{\text{UL}}} \leq \beta, \quad \text{and} \quad \frac{\Lambda}{R_{\text{DL}}} \leq 1 - \beta. \tag{6}$$

Therefore,

$$\frac{\Lambda}{R_{\text{UL}}} + \frac{\Lambda}{R_{\text{DL}}} \leq 1, \tag{7}$$

which is equivalent to

$$\Lambda \leq \frac{1}{\frac{1}{R_{\text{UL}}} + \frac{1}{R_{\text{DL}}}}, \tag{8}$$

revealing the system capacity.

(P1) is a convex optimization problem with respect to β, which can be solved by applying the method of Lagrange multipliers. The Lagrange function is given by

$$\begin{aligned}L(\beta, \nu_1, \nu_2, \nu_3, \nu_4) &= \frac{1}{\beta R_{\text{UL}} - \Lambda} + \frac{1}{(1-\beta) R_{\text{DL}} - \Lambda} - \nu_1 \\ &\cdot (\beta R_{\text{UL}} - \Lambda) - \nu_2((1-\beta) R_{\text{DL}} - \Lambda) - \nu_3 \beta - \nu_4 (1-\beta),\end{aligned} \tag{9}$$

where ν_1, ν_2, ν_3, and ν_4 are the Lagrange multipliers, $\nu_1(\beta R_{\text{UL}} - \Lambda) = 0$, $\nu_2((1-\beta) R_{\text{UL}} - \Lambda) = 0$, $\nu_3 \beta = 0$ and $\nu_4(1-\beta) = 0$. Take derivative with respect to β, we obtain the optimal condition of problem (P1):

$$\begin{aligned}&-\frac{R_{\text{UL}}}{(\beta R_{\text{UL}} - \Lambda)^2} + \frac{R_{\text{DL}}}{((1-\beta) R_{\text{DL}} - \Lambda)^2} \\ &- \nu_1 R_{\text{UL}} + \nu_2 R_{\text{DL}} - \nu_3 + \nu_4 = 0.\end{aligned} \tag{10}$$

Denote by

$$\beta^* = \frac{\sqrt{R_{\text{UL}} R_{\text{DL}}} + \Lambda\left(1 - \sqrt{\frac{R_{\text{UL}}}{R_{\text{DL}}}}\right)}{R_{\text{UL}} + \sqrt{R_{\text{UL}} R_{\text{DL}}}}, \tag{11}$$

which satisfies (10). We can prove β^* is feasible.

$$\begin{aligned}\beta^* - \frac{\Lambda}{R_{\text{UL}}} &= \frac{\sqrt{R_{\text{DL}}} + \frac{\Lambda}{\sqrt{R_{\text{UL}}}} - \frac{\Lambda}{\sqrt{R_{\text{DL}}}}}{\sqrt{R_{\text{UL}}} + \sqrt{R_{\text{DL}}}} - \frac{\Lambda}{R_{\text{UL}}} \\ &= \frac{\sqrt{R_{\text{DL}}}\left[1 + \frac{\Lambda}{\sqrt{R_{\text{UL}} R_{\text{DL}}}} - \frac{\Lambda}{R_{\text{DL}}} - \frac{\Lambda}{\sqrt{R_{\text{UL}} R_{\text{DL}}}} - \frac{\Lambda}{R_{\text{UL}}}\right]}{\sqrt{R_{\text{UL}}} + \sqrt{R_{\text{DL}}}} \\ &= \frac{\sqrt{R_{\text{DL}}}\left[1 - \frac{\Lambda}{R_{\text{DL}}} - \frac{\Lambda}{R_{\text{UL}}}\right]}{\sqrt{R_{\text{UL}}} + \sqrt{R_{\text{DL}}}} \geq 0,\end{aligned} \tag{12}$$

according to (7). Thus, constraint (3 b) holds. In addition,

$$
\begin{aligned}
&\beta^* - 1 + \frac{\Lambda}{R_{\mathrm{DL}}}\\
&= \frac{-\sqrt{R_{\mathrm{UL}}} + \frac{\Lambda}{\sqrt{R_{\mathrm{UL}}}} - \frac{\Lambda}{\sqrt{R_{\mathrm{DL}}}} + \frac{\sqrt{R_{\mathrm{UL}}}}{R_{\mathrm{DL}}}\Lambda + \frac{\Lambda}{\sqrt{R_{\mathrm{DL}}}}}{\sqrt{R_{\mathrm{UL}}} + \sqrt{R_{\mathrm{DL}}}}\\
&= \frac{\sqrt{R_{\mathrm{UL}}}\left[\frac{\Lambda}{R_{\mathrm{UL}}} + \frac{\Lambda}{R_{\mathrm{DL}}} - 1\right]}{\sqrt{R_{\mathrm{UL}}} + \sqrt{R_{\mathrm{DL}}}} \leq 0,
\end{aligned}
\tag{13}
$$

and constraint (3 c) holds. Therefore, β^* is the optimal solution to (P1). Substituting β^* into (2), the minimal average latency can be obtained, given as Theorem 1. □

According to the conventional pull-based NDN scheme, the AoI of vehicle-received content includes two parts: (1) the time of uplink transmission, and (2) the dwelling time at the downlink channel including queueing and transmission[2]. Thus, the average AoI at vehicle side is:

$$
\begin{aligned}
\bar{A}_{\mathrm{con}} &= \frac{1}{\beta^* R_{\mathrm{UL}}} + \frac{1}{R_{\mathrm{DL}}(1-\beta^*) - \Lambda}\\
&= \frac{1+\sqrt{\frac{R_{\mathrm{DL}}}{R_{\mathrm{UL}}}}}{\sqrt{R_{\mathrm{UL}}R_{\mathrm{DL}}}+\Lambda\left(1-\sqrt{\frac{R_{\mathrm{UL}}}{R_{\mathrm{DL}}}}\right)} + \frac{1+\sqrt{\frac{R_{\mathrm{UL}}}{R_{\mathrm{DL}}}}}{\sqrt{R_{\mathrm{UL}}R_{\mathrm{DL}}}-\Lambda\left(\sqrt{\frac{R_{\mathrm{UL}}}{R_{\mathrm{DL}}}}+\sqrt{\frac{R_{\mathrm{DL}}}{R_{\mathrm{UL}}}}\right)}.
\end{aligned}
\tag{14}
$$

The average AoI increases with traffic arrival rate, which can be proved by taking derivative of $\bar{A}_{\mathrm{con}}$ with respect to Λ. This is reasonable as the queue length of the downlink channel increases with traffic load, which is also responsible for the staleness of information.

Corollary 1. *According to the derived capacity, service latency and AoI, the conventional scheme has following features regarding system performance. (1) The capacity depends on the normalized service rates of both uplink and downlink channels; (2) The average AoI and latency both increase with traffic load; and (3) The channel unsymmetry degrades the network capacity and service latency.*

4.2 RSU-Centric Scheme Analysis

Under the RSUC scheme, vehicles obtain the requested content through one-hop downlink transmission. The average service latency is thus given by

$$
\bar{D}_{\mathrm{RC}} = \frac{1}{(1-\beta)R_{\mathrm{DL}} - \Lambda},
\tag{15}
$$

which increases with traffic load and bandwidth splitting ratio β.

The analysis of average AoI is more challenging. From the RSU side, the content update of a typical item s can be modeled as a renewal process. The epoch happens when item s is updated in the RSU cache. Denote by T_{update} the update interval, i.e., the time duration between two successive content updates of the same item, given by

$$
T_{\mathrm{update}} = \sum_{s=1}^{S} t_s,
\tag{16}
$$

where t_s is the uplink transmission time when updating item s. $t_1, t_2, \ldots, t_S$ are independent and identically distributed random variables following exponential distribution of $1/\beta R_{\mathrm{UL}}$. Thus, the update cycle T_{update} follows Erlang-S distribution, and

$$
\begin{aligned}
&\mathbb{E}\left[T_{\mathrm{update}}\right] = \frac{S}{\beta R_{\mathrm{UL}}},\\
&\mathbb{E}\left[T^2_{\mathrm{update}}\right] - \mathbb{E}^2\left[T_{\mathrm{update}}\right] = \frac{S}{(\beta R_{\mathrm{UL}})^2}.
\end{aligned}
\tag{17}
$$

Suppose a vehicle requests item s and the downlink transmission begins at time t. The AoI of the received content is the summation of two parts: (1) the AoI of item s in the RSU cache at time t, and (2) the downlink transmission time for content delivery. For the RSU cache, the AoI of item s at time t also includes two parts: (1) the uplink transmission time to update source s at the last epoch, and (2) the spent time, i.e., the time between t and the last epoch. Denote by T_{b} the spent time, given by

$$
\mathbb{E}[T_{\mathrm{b}}] = \frac{\mathbb{E}\left[T^2_{\mathrm{update}}\right]}{2\mathbb{E}\left[T_{\mathrm{update}}\right]} = \frac{1+S}{2\beta R_{\mathrm{UL}}},
\tag{18}
$$

according to the theories of random process. Denote by T_{UL} and T_{DL} the uplink and downlink transmission time, respectively. Thus, the average AoI of vehicle received content can be obtained:

$$
\begin{aligned}
\bar{A}_{\mathrm{RC}} &= \mathbb{E}[T_{\mathrm{UL}}] + \mathbb{E}[T_{\mathrm{b}}] + \mathbb{E}[T_{\mathrm{DL}}],\\
&= \frac{1}{\beta R_{\mathrm{UL}}} + \frac{1+S}{2\beta R_{\mathrm{UL}}} + \frac{1}{(1-\beta)R_{\mathrm{DL}}}\\
&= \frac{S+3}{2\beta R_{\mathrm{UL}}} + \frac{1}{(1-\beta)R_{\mathrm{DL}}}
\end{aligned}
\tag{19}
$$

which is a convex function with respect to β.

Theorem 2. *Under the RSUC scheme, the AoI and service latency have a trade-off if the ratio of bandwidth allocated to the uplink is no larger than a certain threshold:*

$$
\beta \leq \frac{1}{\sqrt{\frac{2R_{\mathrm{UL}}}{(S+3)R_{\mathrm{DL}}}} + 1}.
\tag{20}
$$

Otherwise, the AoI and service latency both increase with β.

Proof. By taking the first- and second-order derivatives of Eq. (19) with respect to β, we can prove that the average AoI is a convex function of β, with global minimum of $1/\left(\sqrt{\frac{2R_{\mathrm{UL}}}{(S+3)R_{\mathrm{DL}}}} + 1\right)$. According to Eq. (15), the service latency can be proved to increase with β in a monotone manner. Theorem 2 is thus proved. □

Different from the intuition, Theorem 2 indicates that content freshness and service latency do not always have a trade-off. In particular, frequently updating the cached can even degrade the freshness of vehicle-received content, in addition to increasing the service latency. The reason is as

[2] The uplink queuing time does not influence the AoI, since the RSU fetches the new content version when starting to serve the user.

follows. When the bandwidth allocated to uplink is small (i.e., small β), increasing β can significantly improve the freshness of cached contents at the RSU, at the cost of reducing the downlink transmission rate. Thus, the AoI and service latency show a trade-off relationship with respect to bandwidth splitting ratio β. When sufficient bandwidth is allocated for content update (i.e., large β), further increasing β will no longer improve the freshness of cached contents due to marginal gain effect, whereas the extremely low transmission rate in the downlink will stale the contents delivered to vehicles.

Corollary 2. *Under the RSU-centric scheme, the bandwidth splitting ratio should be adjusted to balance the AoI and service latency, in the range of* $\left[0, \frac{1}{\sqrt{\frac{2R_{\rm UL}}{(S+3)R_{\rm DL}}+1}}\right)$.

The threshold of β can be interpreted as saturate point, which indicates the bandwidth allocated to the uplink is sufficient to maintain content freshness. For networks with low uplink transmission rates or large number of sources, the threshold is high and more bandwidth can be allocated to the uplink for content update.

4.3 Request-Adaptive Scheme Analysis

The service process of request-adaptive scheme can be modeled by a M/G/1 queue with the arrival rate of Λ. Denote by X_1 and X_2 the uplink and downlink transmission time following exponential distributions of $1/R_{\rm UL}$ and $1/R_{\rm DL}$, respectively[3]. Denote by I a 0-1 indicator showing if the content update is triggered. Thus, the service time $X = X_2$ if $I = 0$, while $X = X_1 + X_2$ otherwise. As X_1 and X_2 are independent,

$$\begin{aligned}\mathbb{E}[X] &= P(\mathbb{E}[X_1]+\mathbb{E}[X_2]) + (1-P)(\mathbb{E}[X_2]) = \frac{P}{R_{\rm UL}} + \frac{1}{R_{\rm DL}} \\ \mathbb{E}[X^2] &= P\Big(\mathbb{E}[(X_1+X_2)^2]\Big) + (1-P)\big(\mathbb{E}[X_2^2]\big) \\ &= P\mathbb{E}[X_1^2] + 2P\mathbb{E}[X_1]\mathbb{E}[X_2] + \mathbb{E}[X_2^2] \\ &= \frac{2P}{R_{\rm UL}^2} + \frac{2}{R_{\rm DL}^2} + \frac{2P}{R_{\rm UL}R_{\rm DL}},\end{aligned} \tag{21}$$

where $P = \frac{\sum_{s=1}^{S} p_s\lambda_s}{\sum_{s=1}^{S}\lambda_s}$. P is defined as update ratio, denoting the average probability that a content request triggers cache update. Applying the queueing theory, the average service latency of ReA scheme can be obtained:

$$\begin{aligned}\bar{D}_{\rm RA} &= \frac{\Lambda\mathbb{E}[X^2]}{2(1-\Lambda\mathbb{E}[X])} + \mathbb{E}[X] \\ &= \frac{\Lambda\left[\frac{P}{R_{\rm UL}^2} + \frac{1}{R_{\rm DL}^2} + \frac{P}{R_{\rm UL}R_{\rm DL}}\right]}{1-\Lambda\left(\frac{P}{R_{\rm UL}} + \frac{1}{R_{\rm DL}}\right)} + \frac{P}{R_{\rm UL}} + \frac{1}{R_{\rm DL}} \\ &= \frac{\frac{1}{R_{\rm DL}} + \frac{P\Lambda}{R_{\rm UL}^2}}{1-\Lambda\left(\frac{P}{R_{\rm UL}} + \frac{1}{R_{\rm DL}}\right)} + \frac{P}{R_{\rm UL}},\end{aligned} \tag{22}$$

which increases with P (i.e., frequent cache update) and Λ (i.e., heavy traffic load) and decreases with $R_{\rm UL}$ and $R_{\rm DL}$ (i.e., larger service rate).

The network capacity can also be obtained from (22):

$$\hat{\Lambda}_{\rm RA} = \frac{1}{\frac{P}{R_{\rm UL}} + \frac{1}{R_{\rm DL}}}. \tag{23}$$

When $P = 1$, the ReA scheme has the equivalent capacity but lower latency compared with the conventional scheme. In addition, the ReA scheme can always outperform the conventional scheme regarding capacity and latency, since $\hat{\Lambda}_{\rm RA}$ decreases with the update ratio P. Specifically, when $P = 0$, the RSU cache never updates and all bandwidth is used for content delivery (i.e., static caching). The capacity increases to the normalized downlink transmission rate $R_{\rm DL}$, and the service latency decreases to $\frac{1}{R_{\rm DL}-\Lambda}$. When P varies within range (0,1), the network capacity and service latency can be traded by sacrificing content freshness.

The AoI performance can be analyzed in the similar way as the RSUC scheme, where the update process of cached item s can also be modeled as a renewal process. However, the update interval is completely different:

$$T_{{\rm update},s} = \sum_{k=1}^{N_s} T_k, \tag{24}$$

where N_s denotes the number of requests served between two successive updates of item s, and T_k is the time duration between the departures of the kth and $(k-1)$th requests[4]. T_k are i.i.d. random variables following exponential distribution with mean $1/\lambda_s$, applying the properties of M/G/1 queue. Under the ReA scheme, N_s follows geometric distribution: $\mathbb{P}[N_s = n] = p_s(1-p_s)^{n-1}$. Therefore,

$$\mathbb{E}[T_{\rm update,s}] = \sum_{n=1}^{\infty} p_s(1-p_s)^{n-1}\frac{n}{\lambda_s} = \frac{1}{p_s\lambda_s}, \tag{25}$$

$$\mathbb{E}[T_{\rm updat,s}^2] = \sum_{n=1}^{\infty} p_s(1-p_s)^{n-1}\frac{n^2+n}{\lambda_s^2} = \frac{2}{\lambda_s^2 p_s^2}, \tag{26}$$

and the spent time is given by

$$\mathbb{E}[T_{\rm b}] = \frac{\mathbb{E}\left[T_{\rm update}^2\right]}{2\mathbb{E}\left[T_{\rm update}\right]} = \frac{1}{\lambda_s p_s}. \tag{27}$$

If the request for item s triggers update, the average AoI of vehicle received content is given by $\frac{1}{R_{\rm UL}} + \frac{1}{R_{\rm DL}}$. Otherwise, the average AoI equals to $\mathbb{E}[T_{\rm b}] + \frac{1}{R_{\rm DL}}$. As the request triggers update with probability p_s, the overall average AoI of vehicle received content is given by

$$\begin{aligned}\bar{A}_{{\rm RA},s} &= p_s c\left(\frac{1}{R_{\rm UL}} + \frac{1}{R_{\rm DL}}\right) + (1-p_s)\left(\mathbb{E}[T_{\rm b}] + \frac{1}{R_{\rm DL}}\right) \\ &= \frac{p_s}{R_{\rm UL}} + \frac{1}{R_{\rm DL}} + \frac{1-p_{\rm s}}{p_s\lambda_s}.\end{aligned} \tag{28}$$

Based on the derived analytical results, the trade-off between AoI and service latency can be obtained.

[3] Under the ReA scheme, the uplink and downlink share the spectrum resource in the time domain without bandwidth splitting.

[4] The 0th request departs at time 0, without losing generality.

Theorem 3. *Under the ReA scheme, the AoI and service latency always have a trade-off if $R_{\mathrm{UL}} \geq \lambda_s$ for $s = 1, 2, \ldots, S$. Otherwise, the trade-off only holds in the region of $p_s \leq \sqrt{R_{\mathrm{UL}}/\lambda_s}$, $\forall s$.*

Proof. The average AoI is a convex function of p_s with a global minimum of $\sqrt{R_{\mathrm{UL}}/\lambda_s}$. If $R_{\mathrm{UL}} \geq \lambda_s$, the average AoI decreases with $p_s \in [0, 1]$. As the average latency increases with p_s, the average AoI and latency have a trade-off relationship. □

Notice that the content freshness and service latency cannot be traded off if the uplink is overloaded, i.e., $R_{\mathrm{UL}} < \lambda_s$. This may happen due to non-ideal uplink transmissions, e.g., low-power sensor nodes. In this case, restraining the update frequency can enhance both service latency and content freshness, and the update probability should not exceed $\sqrt{R_{\mathrm{UL}}/\lambda_s}$. On the contrary, the freshness and latency can always be traded off if $R_{\mathrm{UL}} \geq \lambda_s$. Here are two extreme cases.

Case 1. As $p_s \to 0$ (i.e., the static caching without update), $\bar{A}_{\mathrm{RA},s} \to \infty$. All resources are utilized for content delivery, which achieves the minimal service latency $\frac{1}{R_{\mathrm{DL}} - \Lambda}$.

Case 2. When $p_s = 1$, all requests trigger update. The average service latency achieves the maximum, given by

$$\frac{\frac{1}{R_{\mathrm{DL}}} + \frac{1}{R_{\mathrm{UL}}} - \frac{\Lambda}{R_{\mathrm{DL}} R_{\mathrm{UL}}}}{1 - \Lambda\left(\frac{1}{R_{\mathrm{UL}}} + \frac{1}{R_{\mathrm{DL}}}\right)}. \tag{29}$$

The corresponding average AoI is given by $1/R_{\mathrm{UL}} + 1/R_{\mathrm{DL}}$, which is the limit of content freshness that can be achieved.

5 RSUC and ReA Scheme Optimization

In this section, the RSUC and ReA schemes are further optimized to enhance the freshness and latency performances simultaneously.

5.1 RSUC Scheme Optimization

Based on Eqs. (15) and (19), the RSUC optimization problem can be formulated:

$$(\mathrm{P2})\ \min_\beta \quad \frac{1}{(1-\beta)R_{\mathrm{DL}} - \Lambda} + W_{\mathrm{A}}\left[\frac{S+3}{2\beta R_{\mathrm{UL}}} + \frac{1}{(1-\beta)R_{\mathrm{DL}}}\right], \tag{30a}$$

$$s.t.\ (1-\beta)R_{\mathrm{DL}} \geq \Lambda, \tag{30b}$$

$$0 \leq \beta \leq 1, \tag{30c}$$

where W_{A} is a weight factor, indicating the importance of AoI compared with service latency. The objective is to minimize both the latency and AoI. The first constraint (30) guarantees that the downlink is not overloaded. Problem (P2) is a convex optimization problem, which can be addressed by the method of Lagrange multipliers. The optimal condition is given by:

$$\frac{1}{\left(\frac{1-\frac{\Lambda}{R_{\mathrm{DL}}}}{\beta^*} - 1\right)^2} + \frac{W_{\mathrm{A}}}{\left(\frac{1}{\beta^*} - 1\right)^2} = \frac{(S+3)W_{\mathrm{A}} R_{\mathrm{DL}}}{2R_{\mathrm{UL}}}. \tag{31}$$

Note that $0 \leq \beta^* \leq 1 - \frac{\Lambda}{R_{\mathrm{DL}}}$, based on the constraint (30 b). Accordingly, the left part of (31) increases with β^*, varying in range of $[0, \infty)$. Therefore, (31) has a unique solution, which can be obtained by dichotomy searching.

The optimal bandwidth splitting ratio β^* achieves the Pareto-optimality of AoI and latency, which further depends on system parameters. In specific, β^* increases with S but decreases with R_{UL}. This result indicates that more resources are consumed to maintain the freshness of more content items, or in case of bad uplink channel conditions. Furthermore, β^* increases with the weight factor W_{A}, revealing the optimal AoI-latency trade-off.

Problem (P2) mainly applies to elastic AoI requirements. For the applications with strict average AoI requirements, the problem can be revised as follows:

$$(\mathrm{P3})\ \min_\beta \quad \frac{1}{(1-\beta)R_{\mathrm{DL}} - \Lambda}, \tag{32a}$$

$$s.t.\ (1-\beta)R_{\mathrm{DL}} \geq \Lambda, \tag{32b}$$

$$\frac{S+3}{2\beta R_{\mathrm{UL}}} + \frac{1}{(1-\beta)R_{\mathrm{DL}}} \leq \hat{A}_{\mathrm{RC}}, \tag{32c}$$

$$0 \leq \beta \leq 1, \tag{32d}$$

where $\hat{A}_{\mathrm{RC}}$ is the threshold for average AoI.

Theorem 4. *If the traffic load satisfies $\Lambda < \frac{R_{\mathrm{DL}}}{1+\sqrt{\frac{(S+3)R_{\mathrm{DL}}}{2R_{\mathrm{UL}}}}}$, the average AoI achieved by RSUC scheme satisfies*

$$\hat{A}_{\mathrm{RC}} \geq \left(\frac{1}{\sqrt{R_{\mathrm{DL}}}} + \sqrt{\frac{S+3}{2R_{\mathrm{UL}}}}\right)^2. \tag{33}$$

The equality holds if the bandwidth splitting ratio is set to

$$\hat{\beta} = 1 - \frac{1}{1 + \sqrt{\frac{(S+3)R_{\mathrm{DL}}}{2R_{\mathrm{UL}}}}}. \tag{34}$$

Proof. According to the constraint (32 c), we have

$$\frac{S+3}{2\beta R_{\mathrm{UL}}} \leq \hat{A}_{\mathrm{RC}}\beta - \frac{\beta}{(1-\beta)R_{\mathrm{DL}}} \tag{35}$$

where the right part can be rewritten as

$$-\hat{A}_{\mathrm{RC}}(1-\beta) - \frac{1}{(1-\beta)R_{\mathrm{DL}}} + \hat{A}_{\mathrm{RC}} + \frac{1}{R_{\mathrm{DL}}}, \tag{36a}$$

$$\leq -2\sqrt{\frac{\hat{A}_{\mathrm{RC}}}{R_{\mathrm{DL}}}} + \hat{A}_{\mathrm{RC}} + \frac{1}{R_{\mathrm{DL}}} \tag{36b}$$

$$= \left(\sqrt{\hat{A}_{\mathrm{RC}}} - \frac{1}{\sqrt{R_{\mathrm{DL}}}}\right)^2, \tag{36c}$$

and the equality of (36 b) holds if $1 - \beta = \sqrt{\frac{1}{\hat{A}_{\mathrm{RC}} R_{\mathrm{DL}}}}$. According to the form of AoI, $\hat{A}_{\mathrm{RC}} > \frac{1}{R_{\mathrm{DL}}}$. Combining (35) and (36), we have

$$\sqrt{\hat{A}_{\mathrm{RC}}} - \frac{1}{\sqrt{R_{\mathrm{DL}}}} - \sqrt{\frac{S+3}{2R_{\mathrm{UL}}}} \geq 0, \tag{37}$$

proving (33) in Theorem 3. Take the equality of (36), the equality of (33) holds. Substitute (33) into $1 - \beta = \sqrt{\frac{1}{\hat{A}_{\mathrm{RC}} R_{\mathrm{DL}}}}$, and we obtain (34). Note that $0 < \hat{\beta} < 1$, and

$(1-\hat{\beta})R_{\rm DL} \geq \Lambda$ under the condition of Theorem 4. Thus, $\hat{\beta}$ is feasible to (P3) and Theorem 4 is thus proved. □

Theorem 4 reveals the limit of content freshness that can be achieved by the RSUC scheme. In specific, the content freshness degrades with the number of content items. Furthermore, the content freshness is more sensitive to the uplink channel condition, especially with large number of content items. In addition, both the vehicular network capacity and service latency have trade-offs with the AoI constraint, as the minimal value of β satisfying constraint (32 c) decreases with $\hat{A}_{\rm RC}$. Here are two cases to show capacity-freshness trade-off. Under the most strict requirement of freshness (i.e., equality of (33)), the network capacity is

$$\hat{\Lambda}_{\rm RC} = \frac{R_{\rm DL}}{1+\sqrt{\frac{(S+3)R_{\rm DL}}{2R_{\rm UL}}}}, \tag{38}$$

with (34) in Theorem 4. On the contrary, without content freshness requirement (i.e., $\hat{A}_{\rm RC} \to \infty$), the vehicular network capacity is given by $\hat{\Lambda}_{\rm RC} \to R_{\rm DL}$ with $\hat{\beta} = 0$. Thus, we can use the RSUC scheme to effectively improve vehicular network capacity in a wider range by sacrificing the content freshness, if there are large number of source nodes or the uplink channel suffers from deep fading. The AoI-latency trade-off can be analyzed similarly.

5.2 AoI and Latency Trade-off Under ReA Scheme

The ReA optimization problem can be formulated as follows[5]:

$$\text{(P4)} \quad \min_{p_s} \quad \frac{\sum_{s=1}^{S} p_s\lambda_s}{\sum_{s=1}^{S}\lambda_s} \tag{39a}$$

$$s.t. \quad \frac{\sum_{s=1}^{S}\lambda_s \bar{A}_{{\rm RA},s}}{\sum_{s=1}^{S}\lambda_s} \leq \hat{A}_{\rm RA}, \tag{39b}$$

$$0 \leq p_s \leq 1, \quad s = 1,2,\ldots,S, \tag{39c}$$

where the objective is to minimize the overall content update ratio, constraint (39 b) guarantees that the system-level average AoI not to exceed the threshold $\hat{A}_{\rm RA}$. As the average service latency increases with the content update ratio, (P4) minimizes the service latency under the given AoI requirement.

Substitute (28) into (39 b), we have

$$\sum_{s=1}^{S}\left[\frac{\lambda_s p_s}{R_{\rm UL}} + \frac{1}{p_s}\right] \leq S + \Lambda\left(\hat{A}_{\rm RA} - \frac{1}{R_{\rm DL}}\right). \tag{40}$$

As (P4) is a convex optimization problem, we can apply Lagrange method to find the solution. The optimality condition is given by

$$\frac{\lambda_s}{\Lambda} + \frac{\nu_0\lambda_s}{R_{\rm UL}} - \frac{\nu_0}{p_s^2} - \nu_1 + \nu_2 = 0, \tag{41}$$

where ν_0, ν_1, and ν_2 are the Lagrange multipliers. Thus,

$$p_s = \frac{1}{\sqrt{\frac{\lambda_s}{\Lambda\nu_0} + \frac{\lambda_s}{R_{\rm UL}} - \frac{\nu_1}{\nu_0} + \frac{\nu_2}{\nu_0}}}. \tag{42}$$

Suppose $0 < p_s < 1$ and $\nu_1 = \nu_2 = 0$. Substituting (42) into (40) and taking equality, we can obtain

$$2\sqrt{\frac{1}{\nu_0\Lambda} + \frac{1}{R_{\rm UL}}} = \left[Y \pm \sqrt{Y^2 - \frac{4}{R_{\rm UL}}}\right], \tag{43}$$

where

$$Y = \frac{S + \Lambda\left(\hat{A}_{\rm RA} - \frac{1}{R_{\rm DL}}\right)}{\sum_{s=1}^{S}\sqrt{\lambda_s}}, \tag{44}$$

for notation simplicity. Note that p_s should take the smaller value which satisfying the equality of (40). Thus, ± should set to + in (43), and

$$\hat{p}_s = \frac{2}{\sqrt{\lambda_s}\left(Y + \sqrt{Y^2 - \frac{4}{R_{\rm UL}}}\right)}. \tag{45}$$

Accordingly, an iterative algorithm can be designed to solve (P4). Denote by $\mathcal{S}_{\rm sub} = \{s|\hat{p}_s \geq 1\}$. If $\mathcal{S}_{\rm sub} = \emptyset$, the optimal solution $p_s^* = \hat{p}_s$ for $s \in \mathcal{S}$, and (P4) is solved. Otherwise, $p_s^* = 1$ for $s \in \mathcal{S}_{\rm sub}$. Set $\mathcal{S} = \mathcal{S} - \mathcal{S}_{\rm sub}$, substitute it into (P4), and calculate (45) in the new iteration.

6 Simulation and Numerical Results

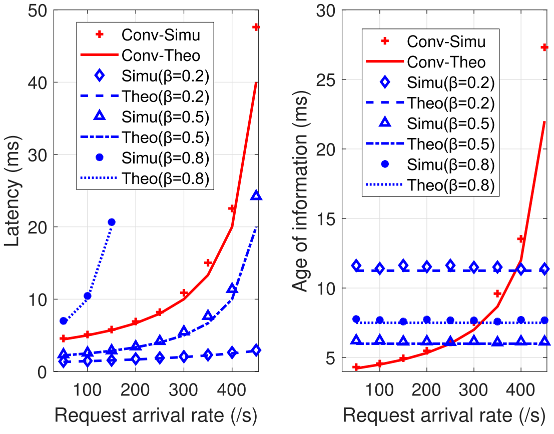
This section conducts system-level simulations to validate the analytical results of AoI and service latency under the RSUC and ReA schemes, by implementing the OMNeT++ simulation platform. In addition, the RSUC and ReA schemes are compared in terms of AoI and latency, under different system parameter settings. Furthermore, the service schemes are evaluated under a practical scenario, where SUMO simulates the real-trace vehicle mobility.

6.1 Analytical Results Validation

The analytical results of RSUC schemes are validated through simulations in case of a single producer generating one item, as shown in Figs. 5. The content size L is set to 3 KB, the normalized service rate for cache update and content delivery are set to $R_{\rm UL} = 1000$ and $R_{\rm DL} = 1000$ content items per second, respectively, while the content request arrival rate Λ varies to reflect different traffic loads. Under the conventional pull-based NDN scheme, the optimal bandwidth splitting is $\beta = 0.5$, due to the symmetric channel gains of producers and vehicles. The simulation results are obtained through OMNeT++ simulator, where the vehicle content requests and transmission time are generated randomly based on the Monte Carlo method. The analytical results are calculated based on Eqs. (2) and (14). For the RSUC scheme, the analytical results are based on the calculations of Eqs. (15) and (19), respectively, considering different bandwidth splitting ratio (β) settings for comparison.

Figs. 5a and 5b show that the simulation results of both latency and AoI are quite close to the analytical ones under

[5] The optimization under elastic AoI requirements is omitted due to page limit. (P4) can provide more insights into system performance.

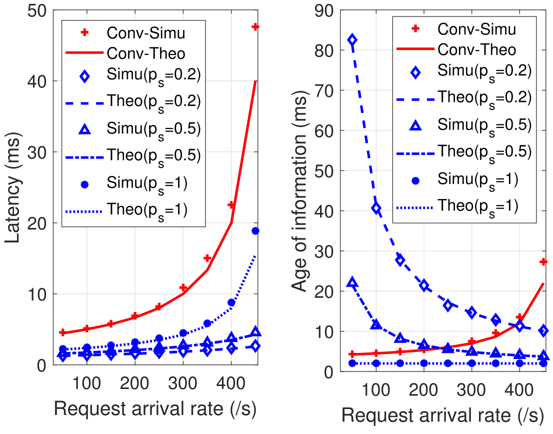


(a) (b)

Fig. 5. Analytical results validation of the RSUC scheme, with conventional NDN pull-based scheme as baseline, (a) average service latency, (b) average age of information.

the conventional NDN and RSUC schemes, which validates the theoretical analysis. Under the conventional pull-based NDN scheme, both the service latency and AoI increase with the content request arrival rate super-linearly. Under the RSUC scheme, the service latency increases while the AoI remains static as the request arrival rate varies for the given bandwidth splitting ratio β. This is reasonable since the RSUC scheme decouples the cache update and content delivery. In comparison, the RSUC scheme can perform better than the conventional NDN scheme in terms of both latency and AoI when the network is heavily loaded (eg., $\beta = 0.2$ or $\beta = 0.5$). The reason is that RSUC reduces cache update frequency, and thus relieves traffic congestion. Furthermore, the service latency and AoI does not always trade off with respect to β. In specific, $\beta = 0.5$ outperforms $\beta = 0.8$ in both latency and AoI. Thus, the bandwidth splitting ratio should not be too large, which is consistent with Theorem 2.

The analytical results of ReA scheme are also validated with different update probability, as shown in Fig. 6. The analytical results of latency and AoI are calculated based on Eqs. (22) and (28), respectively. The simulation results are close to the analytical ones. Under the ReA scheme, the service latency increases with the request arrival rate, while the AoI decreases. Therefore, the ReA scheme is more advantageous in the case of heavy loads, compared with the conventional NDN scheme. Furthermore, the service latency and AoI always shows a trade-off with respect to p_s. In specific, increasing p_s results in higher latency but lower AoI, and $p_s = 1$ can be treated as an extreme case. This result is consistent with Theorem 3.



(a) (b)

Fig. 6. Analytical results validation of the ReA scheme, with conventional NDN pull-based scheme as baseline, (a) average service latency, (b) average age of information.

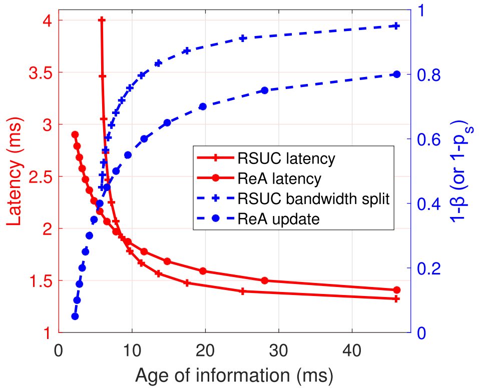


Fig. 7. AoI-Latency trade-off under the two schemes, content request arrival rate $\Lambda = 200$ /s.

6.2 AoI and Latency Trade-Off

Both the analytical and simulation results have demonstrated the existence of AoI-latency trade-off under the RSUC and ReA schemes if the cache update is restrained. For better understanding, we further study the AoI-latency trade-off, by tuning the bandwidth splitting and update probability under the two schemes, respectively. The AoI-latency trade-off is demonstrated by the solid lines in Fig. 7, where the corresponding bandwidth splitting ratio β and update probability p_s are optimized based on problems (P3) and (P4) as the dash lines[6]. The service latency is shown to first decrease and then level off as the AoI increases, under both schemes. Meanwhile, both β and p_s decreases with AoI, indicating that less resources are needed for cache update if the freshness requirement is less strict. The results of Fig. 7 reveal that we can reduce the service latency by sacrificing content freshness, which is especially significant when the AoI requirement is more strict. Regarding the AoI-latency trade-off, the ReA scheme is more beneficial when the AoI requirement is strict (i.e., less than 9 ms), and otherwise the RSUC scheme performs better. This result indicates that we need to choose the appropriate scheme based on the application scenario and system status.

Fig. 8 shows the trade-off between AoI and capacity (i.e., the maximal request arrival rate that can be handled as the service latency goes to infinity) under the two schemes, respectively. As AoI increases, the capacity first increases and then levels off under both schemes. Specifically, the capacity converges to the normalized downlink service rate $R_{\rm DL}$, corresponding to $\beta = 0$. Furthermore, the capacity increases significantly with AoI when the AoI requirement is strict, similar to

[6] $1 - \beta$ and $1 - p_s$ (instead of β and p_s) are shown for better illustration, denoting the resource reserved for content delivery.

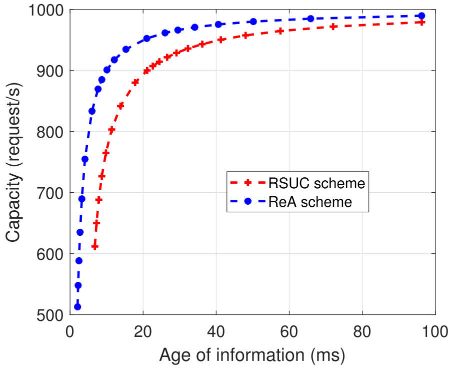


Fig. 8. Capacity-AoI trade-off under the two schemes.

the result of AoI-latency trade-off. In comparison, the ReA scheme outperforms the RSUC scheme in terms of capacity for the given AoI requirement. The reason is that the RSUC should always update the cached content regardless of user arrival, suffering from a higher update cost.

6.3 RSUC-ReA Comparison With Multiple Items

To offer insights into practical ICVN operations, we compare the performance of RSUC and ReA schemes considering the multi-producer scenario, with respect to different system parameters. Fig. 9 shows the AoI-latency trade-off under the two schemes with different number of content items, where each item is requested with an equal probability. As the number of items increases, the trade-off curves of both schemes move rightwards, indicating performance degradation. The reason is that more resources are consumed to maintain the content freshness. In comparison, the ReA scheme can achieve lower latency when the AoI requirement is smaller than a certain threshold. Furthermore, the ReA scheme is more advantageous as the number of items increases.

The RSUC and ReA schemes are also compared considering different item popularity. In specific, the Zipf popularity distribution is considered, where the request probability of the sth most popular item is given by

$$p_s = \frac{1/s^{\theta}}{\sum_{i=1}^{S}(1/i^{\theta})}, \tag{46}$$

where the exponent θ reflects the concentration of requests. The typical value of θ is 0.56, corresponding to the video type services [42]. The comparison of the two schemes in case of Zipf-like popularity is shown in Fig. 10. In specific, the performance of RSUC scheme remains the same whereas the performance of ReA scheme is slightly improved. This result is consistent with the analysis, where the RSUC scheme is not influenced by the request rate of individual items.

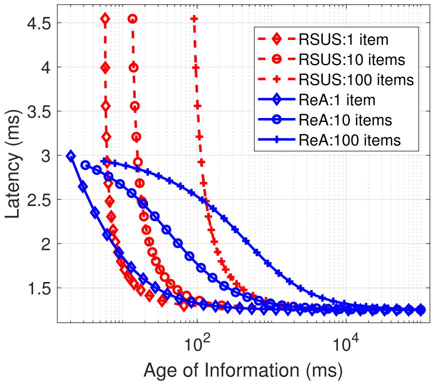


Fig. 10. Comparison of the two schemes, $R_{\text{UL}} = R_{\text{DL}}$=1000 /s, total content request arrival rate $\Lambda = 200$ /s, Zipf item popularity of exponent parameter 0.56.

The influence of request concentration on the ReA scheme is further illustrated, as shown in Fig. 11. The results demonstrate that the performance of the ReA scheme can be improved if the requests are more concentrated, i.e., larger Zipf exponent. Notice that the ReA scheme can adjust update frequency of each item, based on the corresponding arrival rate. This avoids frequent update of unpopular contents, and improves efficiency. Fig. 12 further shows the update frequency and AoI of each item, under the ReA scheme. 10 items are considered with Zipf popularity of 0.56, and the average AoI requirement is 100 ms. The update probability of each item is different, where the less popular contents are updated less frequently. Accordingly, the AoI of less popular contents are higher. This result is consistent with the analysis in [40], where the less popular contents are set with a lower weight when calculating the average AoI. The results of Figs. 11 and 12 reveal the advantages of the ReA scheme when dealing with heterogeneous content requests.

The influence of asymmetric channels are also investigated. In practice, the channel transmission rates from producers to RSUs may be much lower than that from RSUs to

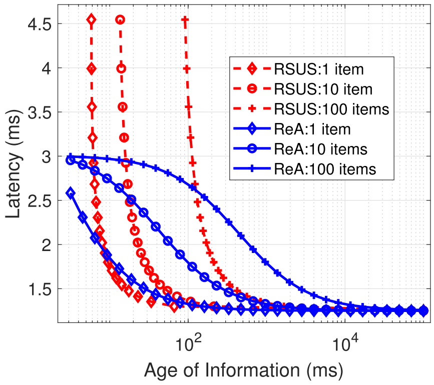


Fig. 9. Comparison of the two schemes, $R_{\text{UL}} = R_{\text{DL}}$=1000 /s, total content request arrival rate $\Lambda = 200$ /s, uniform item popularity.

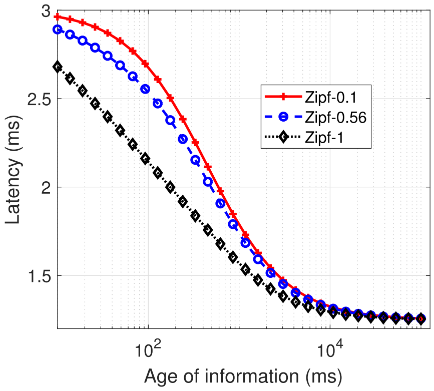


Fig. 11. Influence of content popularity (ReA scheme), $R_{\text{UL}} = R_{\text{DL}}$=1000 /s, request arrival rate $\Lambda = 200$ /s.

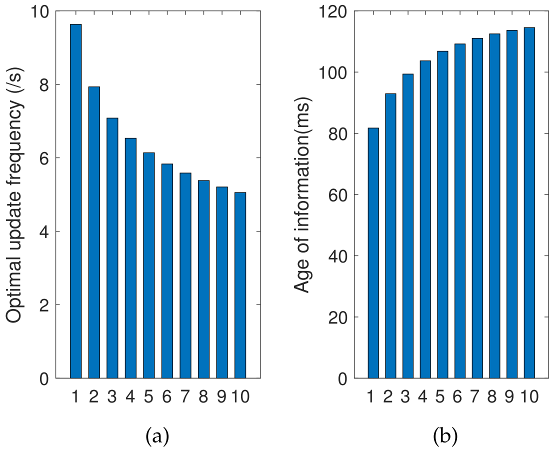


Fig. 12. Optimal update control of individual items, (a) update probability, (b) age of information, average AoI 100 ms, $R_{\rm UL} = R_{\rm DL}$=1000 /s, request arrival rate $\Lambda = 200$ /s, Zipf exponent 0.56.

vehicles, considering the equipped hardware like transceivers and antennas. By setting $R_{\rm UL} = 300$ /s, $R_{\rm DL} = 1000$ /s, we evaluate performance of the two schemes, as shown in Fig. 13. The performance of both schemes degrades significantly, compared with the case of $R_{\rm UL} = R_{\rm DL}$=1000 /s in Fig. 9. Notice that the performance of the RSUC scheme degrades more with larger number of items, and the minimal AoI achieved can increase very fast. By contrast, the minimal AoI achieved by the ReA scheme is not influenced by the number of items, whereas the corresponding latency increases. However, the RUSC scheme completely outperforms the ReA scheme in case of a single item, as shown by the lines with rhombuses.

Based on the comparison of the two schemes, we come to following conclusions: (1) The ReA scheme performs better than the RSUC scheme if the average AoI requirement is smaller than a certain threshold, and vice versa; (2) The ReA scheme is more advantageous with a larger number of items; (3) The ReA scheme adjusts update frequency of individual items based on the popularity, which is more suitable for the heterogeneous application scenarios.

6.4 Real-Trace Simulation

To evaluate the performance of the proposed RSUC and ReA schemes in practical scenarios, we further conduct network-level simulations based on OMNeT++ platform. We consider a RSU deployed at an intersection in the city of Erlangen, Germany, which represents a typical urban scenario. 100 sensor nodes are deployed, which are randomly located within a range of 200 m to the RSU. Each sensor publishes one content item, and the content items are sent to the RSU according to the implemented cache update scheme. The mobility traces of vehicles are generated by the SUMO simulator. Each vehicle raises requests randomly at a certain rate, and all items share equivalent popularity. Accordingly, the requests rate received by the RSU varies with vehicle mobility, as illustrated in Fig. 14. The packet size is set to 3 KB, and the transmission rates of cache update and content delivery are both 24 Mbps, according to the 802.11p standard[7].

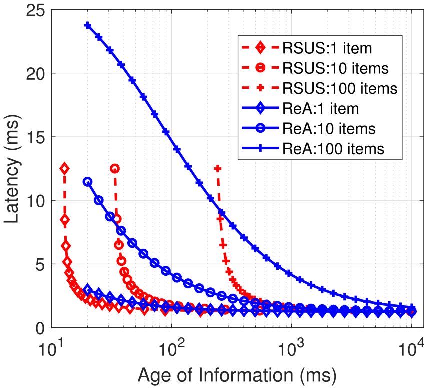


Fig. 13. Latency-AoI trade-off under the two schemes in case of asymmetric channels, $R_{\rm UL} = 300$ /s, $R_{\rm DL} = 1000$ /s, request arrival rate $\Lambda = 200$ /s, uniform content popularity.

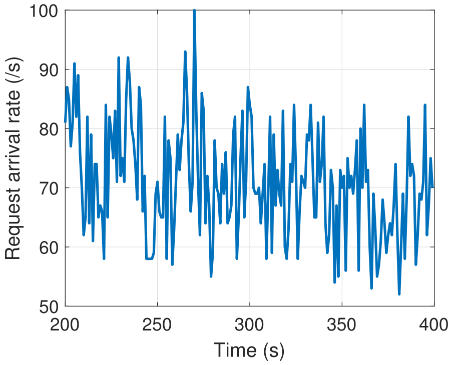
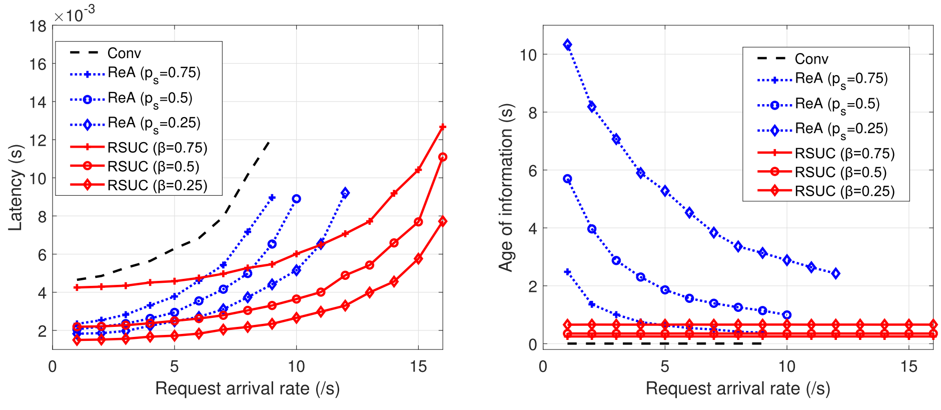


Fig. 14. Illustration of time-varying request arrival rate at the RSU, vehicle mobility trace generated by SUMO.

We implement the proposed RSUC and ReA schemes at the RSU, and the conventional NDN scheme is also conducted as a baseline. The service latency and AoI of each request are recorded, whereby the average value is calculated. Fig. 15 shows the results of different schemes with specified parameter settings, where the x-axis represents the request arrival rate per vehicle. The results show that the conventional NDN scheme achieves the minimal average AoI while the service latency is the highest. Furthermore, the service capacity of the conventional NDN scheme is also the lowest, which is consistent with the analytical results. For example, the network becomes overloaded as the request arrival rate approaches 12 /s under the conventional NDN scheme.

The results demonstrate that both the ReA and RSUC schemes can balance latency and AoI by adjusting the corresponding operational parameters. In comparison, the two schemes can significantly reduce the average latency when the traffic load is high. When the traffic arrival rate is 9 /s, the RSUC scheme with bandwidth splitting ratio of 0.25 can reduce the average latency by nearly 80 percent compared with the conventional NDN scheme, while the ReA scheme can achieve 65 percent reduction. The average AoI will increase to around 0.2 s, which is acceptable for applications whose contents varies at second- or minute-levels. This result indicates the effectiveness to leverage mobile edge caching in ICVNs, while appropriate cache update management schemes is needed.

[7] Different from the analytical model, the transmission rates of uplink and downlink are both constant on the OMNeT++ simulator.

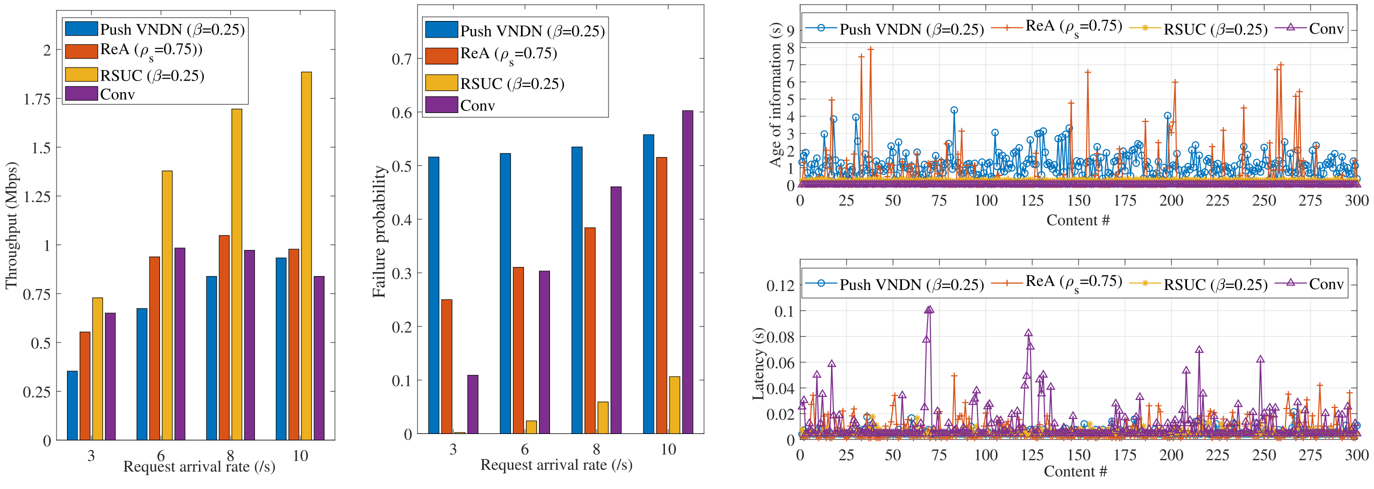


(a) (b)

Fig. 15. Performance evaluation based on SUMO and OMNeT++ platforms: (a) average service latency, and (b) average AoI of user received content items.

We further compare the performance of proposed scheme with the push-based VNDN method [14]. The push-based VNDN allows producers to send generated content items to the RSU prior to requests, which has been proposed to reduce the retrieval latency in case of one producer generating critical event-driven content. As an extension of push-based VNDN, we consider that the multiple producers will push new content versions to the RSU in a contention-based manner. In addition, each producer generates new content versions randomly, and the interval between two updates follows uniform distribution within [0,1] second. The bandwidth splitting ratio is set to 0.25 for both the RSUC and push-based VNDN schemes, and the update probability is set to 0.75 for the ReA scheme, corresponding to the optimal parameter settings in Fig. 15. The latency and AoI requirements are set to 100 ms and 1 s, respectively. Then, we compare the throughput and service failure probability of different schemes under various traffic loads, as shown in Fig. 16. The simulation results of Fig. 16a show that the RSUC scheme can achieve the highest throughput, e.g., around two folds of the ones under other schemes when the traffic load is 10 requests/s. The conventional NDN scheme performs quite well at lower traffic load. However, the throughput of the conventional NDN scheme drops when the request arrival rate exceeds 8 requests/s, indicating that the conventional NDN scheme is overloaded. The ReA scheme outperforms the push-based VNDN scheme in terms of throughput and service failure rate. In comparison with the ReA scheme, the push-based VNDN scheme can achieve lower service latency, whereas the AoI is generally high, as shown in Fig. 16c. The former is because of the favorable one-hop transmission in content retrieval, while the latter is due to the inefficient contention-based cache update.

In general, the RSUC scheme outperforms the ReA scheme according to the simulations. This reason is that the popularity of each item is uniform, and thus the ReA scheme cannot live up to the full potential of fine-grained update frequency. In addition, the update cycle of ReA scheme shows uncertain depending on the random arrival of requests, which may also degrade the freshness performance on average. However, the ReA scheme may outperform the RSUC scheme in case of concentrated requests, which will be studied in future work.



(a) (b) (c)

Fig. 16. Performance comparison with the push-based VNDN method [14], AoI requirement of 1 s and latency requirement of 100 ms, (a) throughput, (b) transmission failures, and (c) AoI and latency of request samples.

6.5 Lessons Learned

Simulations results are summarized to provide insights into practical ICVN cache update and content delivery.

- Both simulation and analytical results demonstrate the existence of AoI-latency trade-off relationship under the RSUC and ReA schemes, if the cache update frequency is restrained.
- The service capacity and AoI requirement also present a trade-off relationship, indicating the costs to maintain content freshness in edge caching.
- The ReA scheme is more beneficial than the RSUC scheme in case of more strict AoI requirement, larger number of items, more concentrated user interests, and lower uplink transmission rates.
- The RSUC scheme can reduce the service latency by 80 percent while guaranteeing the same level of content freshness at high traffic load, in comparison with the conventional pull-based NDN method.
- The RSUC scheme can improve the throughput by around one fold compared with the state-of-the-art push-based VNDN method, when the AoI and latency requirements are set to 1 s and 100 ms, respectively.

7 Conclusion and Future Work

The RSUC and ReA schemes have been proposed to support the dynamic driving-related context information service in ICVNs, through the joint scheduling of cache update and content delivery. Under the proposed schemes, the interplay between service latency and content freshness has been studied. Analytical results have shown that the service latency and content freshness demonstrate a trade-off when the cache update frequency is restrained, but can degrade simultaneously if the RSU cache is frequently updated. In this regard, the RSUC and ReA schemes have been optimized to balance service latency and content freshness on demand. Simulation results have shown that both the RSUC and ReA schemes can effectively improve content freshness and service latency, compared with the conventional pull-based NDN scheme. Furthermore, the ReA scheme outperforms the RSUC scheme in case of strict freshness requirements, heavy traffic loads, larger number of content items, or concentrated content interests. Future studies will further consider the different content sizes and the co-existence of diversified content types. In addition, content items can be also pushed to cache-enabled vehicles based on user interest prediction, which calls for the joint design of content pushing and updating from user side.

Acknowledgments

This work was supported in part by the National Key R&D Program of China under Grant 2019YFB1802803, in part by the Nature Science Foundation of China under Grant 61801011, and in part by Beijing Municipal Natural Science Foundation under Grant L192028 and Grant JZX7Y20190251030001.

References

[1] S. A. A. Shah, E. Ahmed, M. Imran, and S. Zeadally, “5G for vehicular communications,” *IEEE Commun. Magazine*, vol. 56, no. 1, pp. 111–117, Jan. 2018.

[2] D. Grewe, M. Wagner, and H. Frey, “A domain-specific comparison of information-centric networking architectures for connected vehicles,” *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 2372–2388, Thirdquarter 2018.

[3] K. Poularakis and L. Tassiulas, “Code, cache and deliver on the move: A novel caching paradigm in hyper-dense small-cell networks,” *IEEE Trans. Mobile Comput.*, vol. 16, no. 3, pp. 675–687, Mar. 2017.

[4] J. Gao, L. Zhao, and X. Shen, “The study of dynamic caching via state Transition field: The case of time-invariant popularity,” *IEEE Trans. Wireless Commun.*, vol. 18, no. 12, pp. 5924–5937, Dec. 2019.

[5] S. Zhang, P. He, K. Suto, P. Yang, L. Zhao, and X. Shen, “Cooperative edge caching in user-centric clustered mobile networks,” *IEEE Trans. Mobile Comput.*, vol. 17, no. 8, pp. 1791–1805, Aug. 2018.

[6] W. Jiang, G. Feng, and S. Qin, “Optimal cooperative content caching and delivery policy for heterogeneous cellular networks,” *IEEE Trans. Mobile Comput.*, vol. 16, no. 5, pp. 1382–1393, May 2017.

[7] Z. Qu, B. Ye, B. Tang, S. Guo, S. Lu, and W. Zhuang, “Cooperative caching for multiple bitrate videos in small cell edges,” *IEEE Trans. Mobile Comput.*, vol. 19, no. 2, pp. 288–299, Feb. 2019.

[8] W. Zhao, Y. Qin, D. Gao, C. H. Foh, and H. Chao, “An efficient cache strategy in information centric networking vehicle-to-vehicle scenario,” *IEEE Access*, vol. 5, pp. 12 657–12 667, 2017.

[9] Z. Hajiakhondi-Meybodi, J. Abouei, and A. H. F. Raouf, “Cache replacement schemes based on adaptive time window for video on demand services in femtocell networks,” *IEEE Trans. Mobile Comput.*, vol. 18, no. 7, pp. 1476–1487, Jul. 2019.

[10] S. Zhang, J. Chen, F. Lyu, N. Cheng, W. Shi, and X. Shen, “Vehicular communication networks in the automated driving era,” *IEEE Commun. Magazine*, vol. 56, no. 9, pp. 26–32, Sep. 2018.

[11] P. Yang, N. Zhang, S. Zhang, L. Yu, J. Zhang, and X. Shen, “Content popularity prediction towards location-aware mobile edge caching,” *IEEE Trans. Multimedia*, vol. 21, no. 4, pp. 915–929, Apr. 2019.

[12] J. Gao, L. Zhao, and X. Shen, “The study of dynamic caching via state transition field: The case of time-varying popularity,” *IEEE Trans. Wireless Commun.*, vol. 18, no. 12, pp. 5938–5951, Dec. 2019.

[13] J. Gao, S. Zhang, L. Zhao, and X. S. Shen, “The design of dynamic probabilistic caching with time-varying content popularity,” *IEEE Trans. Mobile Comput.*, early access, Jan. 15, 2020, doi: 10.1109/TMC.2020.2967038.

[14] M. F. Majeed, S. H. Ahmed, and M. N. Dailey, “Enabling push-based critical data forwarding in vehicular named data networks,” *IEEE Commun. Letters*, vol. 21, no. 4, pp. 873–876, Apr. 2017.

[15] V. Ortega, F. Bouchmal, and J. F. Monserrat, “Trusted 5G vehicular networks: Blockchains and content-centric networking,” *IEEE Veh. Technol. Magazine*, vol. 13, no. 2, pp. 121–127, Jun. 2018.

[16] E. Kalogeiton and T. Braun, “Infrastructure-assisted communication for NDN-VANETs,” in *Proc. IEEE 19th Int. Symp. World Wireless Mobile Multimedia Netw.*, 2018, pp. 1–10.

[17] A. Baid *et al.*, “Enabling vehicular networking in the MobilityFirst future internet architecture,” in *Proc. IEEE 14th Int. Symp. World Wireless Mobile Multimedia Netw.* 2013, pp. 1–3.

[18] G. Grassi, D. Pesavento, G. Pau, R. Vuyyuru, R. Wakikawa, and L. Zhang, “VANET via named data networking,” in *Proc. IEEE IEEE Conf. Comput. Commun. Workshops*, 2014, pp. 410–415.

[19] J. He, L. Cai, J. Pan, and P. Cheng, “Delay analysis and routing for two-dimensional VANETs using carry-and-forward mechanism,” *IEEE Trans. Mobile Comput.*, vol. 16, no. 7, pp. 1830–1841, Jul. 2017.

[20] G. Grassi, D. Pesavento, G. Pau, L. Zhang, and S. Fdida, “Navigo: Interest forwarding by geolocations in vehicular named data networking,” in *Proc. IEEE 16th Int. Symp. World Wireless Mobile Multimedia Netw.*, 2015, pp. 1–10.

[21] A. Boukerche, R. W. L. Coutinho, and X. Yu, “LISIC: A link stability-based protocol for vehicular information-centric networks,” in *IEEE 14th Int. Conf. Mobile Ad Hoc Sensor Syst.*, 2017, pp. 233–240.

[22] J. Chen, M. Jahanian, and K. K. Ramakrishnan, “Black ice! using information centric networks for timely vehicular safety information dissemination,” in *Proc. IEEE Int. Symp. Local Metropolitan Area Netw.*, 2017, pp. 1–6.

[23] M. Zhang, H. Luo, and H. Zhang, “A survey of caching mechanisms in information-centric networking,” *IEEE Commun. Surv. Tuts.*, vol. 17, no. 3, pp. 1473–1499, Thirdquarter 2015.

[24] F. M. Modesto and A. Boukerche, "An analysis of caching in information-centric vehicular networks," in *Proc. IEEE Int. Conf. Commun.*, 2017, pp. 1–6.

[25] S. Zhang, N. Zhang, X. Fang, P. Yang, and X. Shen, "Self-sustaining caching stations: Toward cost-effective 5G-enabled vehicular networks," *IEEE Commun. Magazine*, vol. 55, no. 11, pp. 202–208, Nov. 2017.

[26] G. Mauri, M. Gerla, F. Bruno, M. Cesana, and G. Verticale, "Optimal content prefetching in NDN vehicle-to-infrastructure scenario," *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2513–2525, Mar. 2017.

[27] S. Zhang, W. Quan, J. Li, W. Shi, P. Yang, and X. Shen, "Air-ground integrated vehicular network slicing with content pushing and caching," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 2114–2127, Sep. 2018.

[28] C. Li, S. Gong, X. Wang, L. Wang, Q. Jiang, and K. Okamura, "Secure and efficient content distribution in crowdsourced vehicular content-centric networking," *IEEE Access*, vol. 6, pp. 5727–5739, 2018.

[29] L. Yao, A. Chen, J. Deng, J. Wang, and G. Wu, "A cooperative caching scheme based on mobility prediction in vehicular content centric networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 5435–5444, Jun. 2018.

[30] S. Kaul, M. Gruteser, V. Rai, and J. Kenney, "Minimizing age of information in vehicular networks," in *Proc. 8th Annu. IEEE Commun. Soc. Conf. Sensor Mesh Ad Hoc Commun. Netw.*, 2011, pp. 350–358.

[31] A. Kosta, N. Pappas, and V. Angelakis, *Age of Information: A New Concept, Metric, and Tool*. Now Foundations and Trends, 2017. [Online]. Available: https://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=8187436

[32] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?," in *Proc. IEEE INFOCOM*, 2012, pp. 2731–2735.

[33] E. Najm and R. Nasser, "Age of information: The gamma awakening," in *Proc. IEEE Int. Symp. Inf. Theory*, 2016, pp. 2574–2578.

[34] R. D. Yates and S. K. Kaul, "The age of information: Real-time status updating by multiple sources," *IEEE Trans. Inf. Theory*, vol. 65, no. 3, pp. 1807–1827, Mar. 2019.

[35] C. Kam, S. Kompella, G. D. Nguyen, and A. Ephremides, "Effect of message transmission path diversity on status age," *IEEE Trans. Inf. Theory*, vol. 62, no. 3, pp. 1360–1374, Mar. 2016.

[36] M. Costa, M. Codreanu, and A. Ephremides, "On the age of information in status update systems with packet management," *IEEE Trans. Inf. Theory*, vol. 62, no. 4, pp. 1897–1910, Apr. 2016.

[37] G. Stamatakis, N. Pappas, and A. Traganitis, "Optimal policies for status update generation in a wireless system with heterogeneous traffic," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 5315–5328, Jun. 2020.

[38] R. D. Yates, M. Tavan, Y. Hu, and D. Raychaudhuri, "Timely cloud gaming," in *Proc. IEEE Conf. Comput. Commun.*, 2017, pp. 1–9.

[39] C. Kam, S. Kompella, G. D. Nguyen, J. E. Wieselthier, and A. Ephremides, "Information freshness and popularity in mobile caching," in *Proc. IEEE Int. Symp. Inf. Theory*, 2017, pp. 136–140.

[40] R. D. Yates, P. Ciblat, A. Yener, and M. Wigger, "Age-optimal constrained cache updating," in *Proc. IEEE Int.l Symp. Inf. Theory*, Jun. 2017, pp. 141–145.

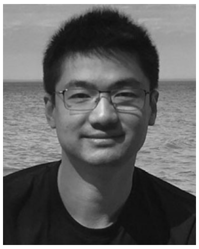
[41] S. Zhang, J. Li, H. Luo, J. Gao, L. Zhao, and X. Shen, "Towards fresh and low-latency content delivery in vehicular networks: An edge caching aspect," in *Proc. 10th Int. Conf. Wireless Commun. Signal Process.*, 2018, pp. 1–6.

[42] P. Gill, M. Arlitt, Z. Li, and A. Mahanti, "Youtube traffic characterization: A view from the edge," in *Proc. 7th ACM SIGCOMM Conf. Internet Meas.*, 2007, pp. 15–28.

Shan Zhang (Member, IEEE) received the PhD degree in electronic engineering from Tsinghua University, Beijing, China, in 2016. She is currently an associate professor at the School of Computer Science and Engineering, Beihang University, Beijing, China. She was a postdoctoral fellow in the Department of Electronical and Computer Engineering, University of Waterloo, Ontario, Canada, from 2016 to 2017. Her research interests include mobile edge computing, wireless network virtualization, and intelligent management. She received the Best Paper Award at the Asia-Pacific Conference on Communication, in 2013. She has been serving as an associate editor for *Peer-to-Peer Networking and Applications*, and a guest editor for *China Communications*.

Junjie Li received the BS degree in flight vehicle propulsion engineering from the Civil Aviation University of China, Tianjin, China, in 2016. He is currently working toward the MS degree from the School of Computer Science and Engineering, Beihang University. His research interests include future internet architecture and UAV networks.

Hongbin Luo (Member, IEEE) received the BS degree from Beihang University, in 1999, and the MS (with honors) and PhD degrees in communications and information science from the University of Electronic Science and Technology of China (UESTC), in June 2004 and March 2007, respectively. He is currently a professor at the School of Computer Science and Engineering, Beihang University. From June 2007 to March 2017, he worked at the School of Electronic and Information Engineering, Beijing Jiaotong University. From September 2009 to September 2010, he was a visiting scholar at the Department of Computer Science, Purdue University. He has authored more than 50 peer-reviewed papers in leading journals (such as *IEEE/ACM Transactions on Networking*, *IEEE Journal on Selected Areas in Communications*) and conference proceedings. In 2014, he won the National Science Fund for Excellent Young Scholars from the National Natural Science Foundation of China (NSFC). His research interests include the wide areas of network technologies including network architecture, routing, and traffic engineering.



Jie Gao (Member, IEEE) received the MSc and PhD degrees in electrical and computer engineering from the University of Alberta, Canada, in 2009 and 2014, respectively. He joined the Department of Electrical and Computer Engineering, Marquette University, Milwaukee, USA, as an assistant professor, in August 2020. Before that, he was a research associate with the University of Waterloo, Waterloo, Canada, from 2019 to 2020 and a postdoc with Ryerson University, Toronto, Canada, from 2017 to 2019. His research interests include machine learning for communications and networking, Internet of things (IoT) and industrial IoT solutions, and cloud and edge computing.

Lian Zhao (Senior Mmeber, IEEE) received the PhD degree from the University of Waterloo, Canada, in 2002. She joined the Department of Electrical, Computer and Biomedical Engineering at Ryerson University, Toronto, Canada, in 2003 and was a professor, in 2014. Her research interests include the areas of wireless communications, resource management, mobile edge computing, caching and communications, and vehicular adhoc networks. She has been selected as an IEEE Communication Society (ComSoc) distinguished lecturer (DL) for 2020 and 2021, received the Best Land Transportation Paper Award from IEEE Vehicular Technology Society in 2016, Best Paper Award from the 2013 International Conference on Wireless Communications and Signal Processing (WCSP) and Best Student Paper Award (with her student) from Chinacom 2011, and the Canada Foundation for Innovation (CFI) New Opportunity Research Award in 2005. She has been serving as an editor for *IEEE Transactions on Vehicular Technology*, *IEEE Transactions on Wireless Communications*, and *IEEE Internet of Things Journal*. She served as a co-chair of Wireless Communication Symposium for IEEE Globecom 2020 and IEEE ICC 2018, co-chair of Communication Theory Symposium for IEEE Globecom 2013, Finance co-chair for IEEE ICASSP 2021. She served as a committee member for NSERC (Natural Science and Engineering Research Council of Canada) Discovery Grants Evaluation Group for Electrical and Computer Engineering 2015 to 2018. She is a licensed professional engineer in the Province of Ontario and a senior member of the IEEE Communication and Vehicular Society.

Xuemin (Sherman) Shen (Fellow, IEEE) received the PhD degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990. He is currently a university professor at the Department of Electrical and Computer Engineering, University of Waterloo, Canada. His research interests include network resource management, wireless network security, Internet of Things, 5G and beyond, and vehicular ad hoc and sensor networks. He is a registered professional engineer of Ontario, Canada, an Engineering Institute of Canada fellow, a Canadian Academy of Engineering fellow, a Royal Society of Canada fellow, a Chinese Academy of Engineering Foreign member, and a distinguished lecturer of the IEEE Vehicular Technology Society and Communications Society. He received the R.A. Fessenden Award, in 2019 from IEEE, Canada, Award of Merit from the Federation of Chinese Canadian Professionals (Ontario), in 2019, James Evans Avant Garde Award, in 2018 from the IEEE Vehicular Technology Society, Joseph LoCicero Award, in 2015 and Education Award, in 2017 from the IEEE Communications Society, and Technical Recognition Award from Wireless Communications Technical Committee (2019) and AHSN Technical Committee (2013). He has also received the Excellent Graduate Supervision Award, in 2006 from the University of Waterloo and the Premier's Research Excellence Award (PREA), in 2003 from the Province of Ontario, Canada. He served as the Technical Program Committee chair/co-chair for IEEE Globecom'16, IEEE Infocom'14, IEEE VTC'10 Fall, IEEE Globecom'07, and the chair for the IEEE Communications Society Technical Committee on Wireless Communications. He is the elected IEEE Communications Society vice president for Technical & Educational Activities, vice president for Publications, Member-at-Large on the Board of Governors, chair of the Distinguished Lecturer Selection Committee, member of IEEE ComSoc fellow Selection Committee. He was/is the editor-in-chief of the *IEEE Internet of Things Journal*, *IEEE Network*, *IET Communications*, and *Peer-to-Peer Networking and Applications*.