

Adaptive Virtual RSU Scheduling for Scalable Coverage under Bidirectional Vehicle Traffic Flow

Fei Chen ^{1,*}, Xiaohong Bi ¹, Ruimin Lyu ¹, Zhongwei Hua ², Yuan Liu ¹ and Xiaoting Zhang ¹

¹ School of Digital Media, Jiangnan University, Wuxi 214122, China; bxhdockerjiangnan@163.com (X.B.); magicbrush.lv@gmail.com (R.L.); lyuan1800@jiangnan.edu.cn (Y.L.); zhangxt6@163.com (X.Z.)

² Applied Technology College, Soochow University, Soochow 215325, China; huorantiancai@163.com

* Correspondence: chenf@jiangnan.edu.cn

Received: 12 May 2017; Accepted: 18 August 2017; Published: 24 August 2017

Abstract: Over the past decades, vehicular ad hoc networks (VANETs) have been a core networking technology to provide drivers and passengers with safety and convenience. As a new emerging technology, the vehicular cloud computing (VCC) can provide cloud services for various data-intensive applications in VANETs, such as multimedia streaming. However, the vehicle mobility and intermittent connectivity present challenges to the large-scale data dissemination with underlying computing and networking architecture. In this paper, we will explore the service scheduling of virtual RSUs for diverse request demands in the dynamic traffic flow in vehicular cloud environment. Specifically, we formulate the RSU allocation problem as maximum service capacity with multiple-source and multiple-destination, and propose a bidirectional RSU allocation strategy. In addition, we formulate the content replication in distributed RSUs as the minimum replication set coverage problem in a two-layer mapping model, and analyze the solutions in different scenarios. Numerical results further prove the superiority of our proposed solution, as well as the scalability to various traffic condition variations.

Keywords: VANETs; online algorithm; distributed solutions

1. Introduction

Large-scale data dissemination, e.g., multimedia streaming applications, has become one of the major tasks of vehicular ad hoc networks (VANETs) [1]. Much effort is required to deal with some issues in vehicular environments, such as the fast changing topology, the shortened intermittent connectivity, the unique set of applications, and the harsh propagation conditions [2–4]. Such distinguishing features demand the design of new networking solutions [5,6].

As an extension of mobile cloud computing, vehicular cloud computing (VCC) is introduced as a new paradigm, which aims to make an efficient use of resources available in vehicles, such as computing, sensing, and communication to provide useful services [7,8]. Unlike the traditional Internet cloud, which is created and maintained by a cloud provider, the vehicle cloud is temporarily created by interconnecting resources available in the vehicles and roadside units (RSUs). The location-based and time-dependent vehicular applications could potentially benefit from the in-network and decentralized data caching and replication mechanisms [9–11]. A resource-constrained RSU, joining the cloud as a front-edge node, can be a good candidate for the service of content delivery. Given the RSUs as the edge nodes along the road, these distributed nodes can constitute a fog computing environment, which can connect the dynamic traffic flow with a remote cloud to facilitate the content delivery in urban areas.

Due to the intermittent connection, content delivery in VCC is particularly susceptible to the time-varying traffic condition, as well as the diverse request demand [12]. In order to improve the efficiency for real-world implementation, we have to consider a dynamic scenario in which the traffic

condition and request demand cannot be predicted for the next time slot. It is different from the static scenario, in which all the statistics can be known in advance and the global optimization can be achieved. In the dynamic environment, the adaptive solutions have to be implemented according to the current conditions for the services in the following time slots. To deal with this problem, infrastructure based networks hold a significant advantage: access points allow optimal scheduling of network resources distribution in a relatively simple manner, at the cost of RSU deployment throughout the intended coverage area [13–16]. Consequently, the efficiency of content delivery strategy is largely influenced by the provisioning location and the replication content of these distributed RSUs.

In this paper, we consider a cloud-based architecture, in which the virtual RSUs (denoted as RSUs for short in the following paper) can act as the front edge nodes of the cloud platform. We will consider the RSU-level cooperation in the dynamic environment, which refers to not only the traffic condition but also the request demand distributions. Specifically, we will explore the service efficiency among multiple RSUs along a road segment with bidirectional vehicle traffic flow. To deal with the vehicle traffic variation, we present a bidirectional allocation algorithm to select the bottleneck with time-varying link capacity. In addition, we further present a two-layer model and explore the replication strategy in a timely manner to update the content replication for the request demand diversity. The contributions of this paper are listed as follows:

1. First, we present a system design for the content delivery service in the urban area, including the RSU allocation and cache replication. The system structure is explored to organize comprehensive factors and analyze the potential connections between them.
2. Second, as we have known, this is the first paper to explore the RSU allocation in a road segment with bidirectional traffic flows. In addition, we extend our analysis about the cache replication from a single road segment scenario to an urban area scenario.
3. Third, the proposed solutions are designed to accommodate the time-varying traffic conditions and request demands. These strategies can also be deployed in the distributed road segments or RSUs for scalable implementation.

The remainder of this paper proceeds as follows. The related works are presented in Section 2. In Section 3, we present the system structure and our adaptive virtual RSU scheduling strategy design. In Section 4, the service capacity with multiple sources and multiple destinations is investigated, and a bidirectional RSU allocation strategy is developed to increase the link capacity along the road. In Section 5, we present a two-layer model for content replication, and analyze the solutions in a special case scenario and general case scenario, respectively. The numerical results are presented in Section 6 to evaluate the performance under different strategies. Finally, Section 7 concludes this paper and discusses potential future directions.

2. Related Work

The content allocation for distributed RSUs in a large scale has been greatly explored in recent years [13–16]. Generally, these works can be divided into two types, including the RSU allocation and cache replication.

RSU allocation refers to the position selection according to the vehicle distribution. For example, Wu et al. [13] proposed a cost-effective strategy for RSU placement in VANETs with the formulation of vehicle population distribution and the vehicle speed. In [14], a connectivity-oriented maximum coverage RSU deployment strategy was proposed to maximize vehicle-to-infrastructure (V2I) communication performance in urban areas. In [15], an RSU placement strategy was proposed to improve the location impacts on the efficiency of vehicular network with minimal cost provisioning. Furthermore, Abdrabou et al. [16] presented an analytical RSU placement framework, which takes into account the randomness of vehicle data traffic and the statistical variation of the disrupted communication channel. However, none of these works have taken into account the bidirectional traffic flow, which is prevalent in modern urban areas. Most previous works use the vehicle density instead of

the traffic flow, and can be considered as static scenarios in which the traffic condition is independent in different time slots. Even though some previous works analyze the single-direction traffic flow, they assume a simplified scenario in which the vehicles will pass by the regions in a sequence, neglecting the impact of traffic flow in the opposite direction. Recently, some works have presented the bidirectional traffic model to analyze the efficiency of vehicle-to-vehicle (V2V) communications in VANET. For example, Liu et al. studied the message delivery delay in VANETs with the impact factors of message delivery distance and density of vehicles based on the bidirectional traffic model in [17]. According to the driving directions and relative positions of vehicles, Wang et al. designed a time slot division policy to assign the vehicle clusters with specific slots in [18]. In [19], an inter-vehicle ad hoc routing metric based on the density and velocities of vehicles was proposed, and the clusters in opposite directions were used as a bridge to reduce the message propagation delay. Lee et al. also analyzed the data delivery delay on a bidirectional road segment with carry-and-forward vehicle communications [20]. Nevertheless, these works mentioned above only investigate the influence of bidirectional traffic on the V2V communications with the assumption that the RSUs are allocated at the ends of a road segment. None of previous research has explored the RSU allocation under the bidirectional traffic flows, which make the problem more complicated. Intuitively, the RSUs should be allocated in the regions with highest vehicle density for a larger coverage. However, it is not the case for the bidirectional traffic flow scenario, in which the region with high traffic density may experience transmission bottleneck in either direction. Therefore, we are motivated to present a comprehensive model for the bidirectional traffic flow and analyze the solutions for the RSU allocation problem.

As to the cache replication, it refers to replica allocation in the fixed RSUs with limited capacity according to various request demands. For example, in Ref. [21], a novel cache invalidation algorithm was proposed to take advantage of the underlying location management scheme to reduce the number of broadcast operations and corresponding query delay. Ding et al. [22] proposed an optimal, a sub-optimal, and a greedy strategy for RSU content replication to minimize the average request delay. However, none of these works have taken into account both the RSU allocation and request popularity during the content replication. In addition, Wang et al. [23] proposed a globally optimal online EV charging strategy, which not only improves energy utilization of the whole system but also prevents charging stations from overloading, which may cause a voltage drop in the power distribution system. In [24], a publish–subscribe based event notification framework was proposed to deliver events to vehicles that subscribe to them within the validity periods of both the subscriptions and the events. Nevertheless, none of these strategies considers the content replication in the dynamically provisioned RSUs with limited storage capacity under diverse request demands. Here, we will present a system design for the RSU service in the urban area, and explore the service efficiency under the impact of comprehensive factors after the RSU allocation.

Leveraging mobile cloud paradigms, VCC has been proposed recently to change the way of network service provisioning in VANETs. Many works use underutilized vehicle resources to form a cloud by aggregating vehicular computing resources. For example, a distributed D-hop cluster formation algorithm was presented to dynamically form vehicle clouds for efficient data delivery in [25]. Feteiha et al. [26] proposed a precoded cooperative transmission technique to extract the underlying rich multipath-Doppler-spatial diversity, which is a relay selection scheme for available relaying vehicles. Kim et al. [27] further presented a prefetch based strategy from remote cloud to RSUs for efficient data dissemination in VCC. Different from previous works, we will explore service provisioning in VCC under the impact of a macro view of traffic conditions and request demand variation.

3. System Structure Overview

We consider a vehicular cloud computing architecture in Figure 1, which includes vehicles in dynamic traffic flow, virtual RSUs distributed in urban areas, and a cloud platform to support on a large scale. Initially, the traffic flow map in urban areas can be achieved through the distributed

traffic monitoring or a third party Internet map service provider. With the traffic flow map, the virtual RSU scheduling can be implemented by cloud platform for a large scale content delivery coverage. The RSUs cache the file items that can be downloaded by the vehicles nearby. There are a large set of files in total, which are all stored in the cloud platform as backups. Meanwhile, the RSU storage capacity is limited, and cache items need to be updated according to the replication strategy. Each RSU receives the requests from the traffic flow in its coverage area, and feeds back the request demand according to its local replication. From the viewpoint of a specific vehicle, its request demand can be fulfilled by the RSUs passing by; otherwise, it will resort to the support of cloud platform or vehicle-to-vehicle communications. The RSU scheduling strategy includes RSU allocation and cache replication. Specifically, the RSU allocation determines where to allocate the RSUs, and the content replication determines what to cache in these allocated RSUs.

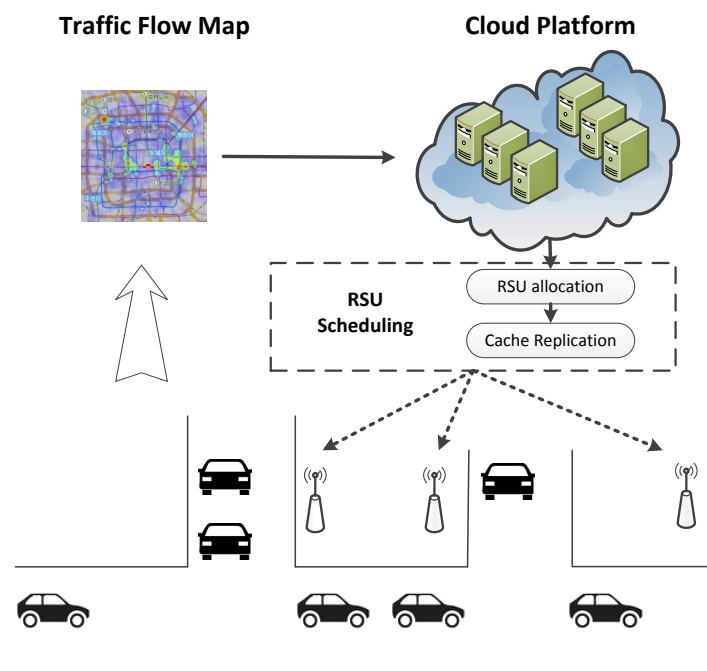


Figure 1. A road segment with a bidirectional road.

RSU Allocation. Given the traffic flow map, RSU allocation will select an RSU distribution to balance the system cost and performance. As we assume a virtual RSU based on network function virtualization (NFV) technology, the RSU can be started or terminated dynamically according to the time-varying traffic flows. Intuitively, provisioning more RSUs would bring higher service coverage, and also leads to higher system cost. Through V2V and V2I communications in VANETs, the RSUs are allocated to facilitate the content delivery under the cost constraints. In other words, given a limited number of RSUs, we need to find an appropriate RSU placement to improve the connectivity of the whole system.

Cache Replication. After the RSUs are allocated, the cache replication strategy refers to the file item management in the local storage of RSUs. As the RSUs usually have limited storage capacity, the content replication needs to be updated according to the various request demands of vehicles passing by. Naively caching the most popular item content can hardly guarantee the efficiency of the whole system, as the RSUs are usually allocated along the road and accessed by the passing-by vehicles in sequence. Therefore, we need to consider both the request demand and RSU distribution, and explore the service scheduling strategy among the neighboring RSUs.

We present the details of the RSU scheduling strategy design procedures in Figure 2, including the RSU allocation and cache replication. Initially, we consider the dynamic environment with time-varying

traffic conditions and various request demands. According to the traffic conditions, the RSU allocation strategy is designed to improve the service capacity with a limited number of RSUs. According to the RSU distribution and the popularity of the request demand, the cache replication is designed to improve the service efficiency with storage capacity constraint.

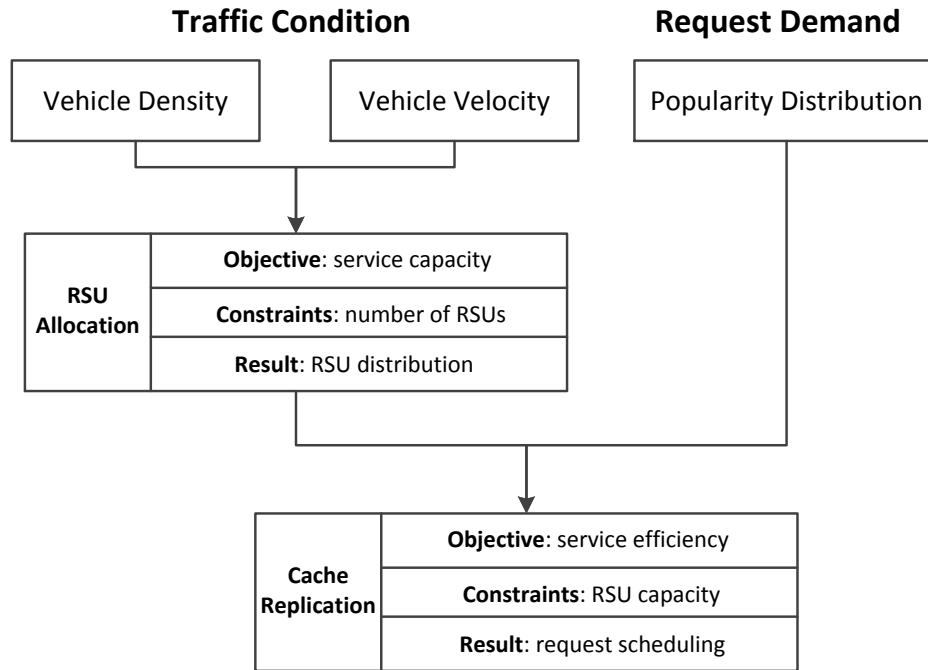


Figure 2. RSU scheduling strategy design.

4. RSU Allocation

As the traffic condition is relatively independent for each region, the global traffic flow can be divided into different road segments. In this section, we will illustrate the RSU allocation in a road segment with bidirectional traffic flow.

4.1. Communication in Bidirectional Traffic Flow

In Figure 3, we consider a road segment with the eastbound flow $V^e \in \mathbb{V}$ and the westbound flow $V^w \in \mathbb{V}$. These flows are independent from each other, in terms of vehicle density and traffic conditions. The road segment can be divided into l different regions. Suppose that we have k RSUs and $l \gg k$ RSUs for content allocation. We suppose that vehicle density in region $i \in \{1, \dots, l\}$ can be represented as $d^e(i)$ for eastbound flow and $d^w(i)$ for westbound flow.

Generally, we consider three types of data communications: (1) Vehicle communications in the same direction represent the inter-vehicle communications with one hop or multiple hops along the traffic flow of same direction. The communication time is determined by the relative speed of vehicles, which can be assumed to be stable for each specific region. Here, we can denote $\tau_s^e(i)$ and $\tau_s^w(i)$ as the communication time for the eastbound flow and the westbound flow in region i ; (2) Vehicle communications in the opposite direction represent the inter-vehicle communications between vehicles driving in the opposite direction. We denote $\tau_o(i)$ as the relative communication time for the opposite traffic flow in region i . Usually, we have $\tau_o \ll \tau_s$, as the relative speed of vehicles in the opposite direction is much less than that in the same direction; (3) Vehicle-to-RSU communications represent the data communications between the RSU and the vehicles passing by. Denote $\tau_r^e(i)$ and $\tau_r^w(i)$ as the vehicle-to-RSU communication time in region i for the eastbound traffic and the westbound traffic. Rather than the relative speed in the previous two types of communications, the vehicle-to-RSU

communication is determined by the absolute vehicle speed, as the allocated RSUs are static. Obviously, we have the relationship between τ_r and τ_o as follows:

$$\frac{1}{\tau_r^e(i)} + \frac{1}{\tau_r^w(i)} \propto \frac{1}{\tau_o(i)}.$$

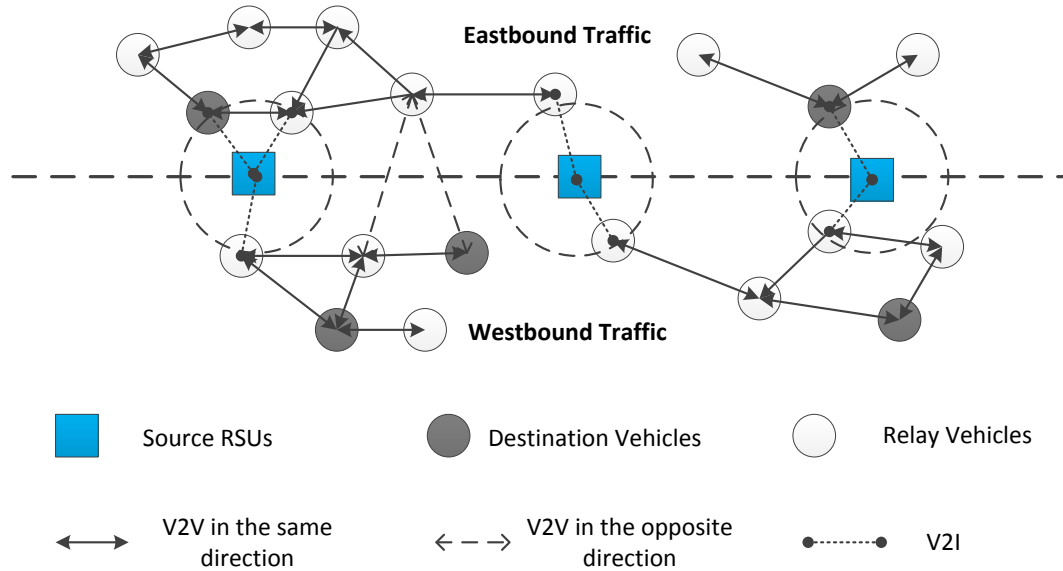


Figure 3. RSU allocation in a constructed graph.

4.2. Service Capacity with Multiple Sources and Multiple Destinations

Given the three types of data communications and the vehicle density, we can have a directed graph G with edge capacities $u(e) \in \mathbb{R}_+$ and $e \in E(G)$ as the estimated communication connectivity between neighbor regions. Each region includes bidirectional traffic flows for content dissemination. There are especially two specific regions as s (i.e., the region allocated with RSUs as the source node) and χ (i.e., the region as the destination node). Our main motivation is to improve the communication capacity from s to χ . Within the quadruple (G, u, s, χ) , we can have a definition as follows:

Definition 1. Given a digraph G with capacities $u(e)$, a flow is a function $f(e) \in \mathbb{R}_+$ with $f(e) \leq u(e)$ for all $e \in E(G)$. The excess of a flow f at $v \in V(G)$ is

$$ex_f(v) := \sum_{e \in \delta^-(v)} f(e) - \sum_{e \in \delta^+(v)} f(e).$$

Now, given a network (G, u, s, χ) , an s - χ -flow is a flow f from source node s to destination node χ , and we can define the value of an s - χ -flow f as $|f(s, \chi)| = ex_f(\chi)$.

Definition 2. Given the definition of an s - χ -flow for $\chi \in X$, we can further define a flow set as $F(S, X)$ with a value $|F(S, X)| = \min\{ex_f(\chi) \mid \chi \in X \text{ and } s \in S\}$.

Accordingly, we can formulate the problem as follows:

$$\begin{aligned} \text{Max.} \quad & |F(S, X)|, \\ \text{s.t.} \quad & ex_f(S) = \sum_{s \in S} ex_f(s) \leq 0, \\ & \forall v \in V(G) \setminus \{S, X\} \quad ex_f(v) = 0. \end{aligned}$$

In this problem, we are motivated to increase the service connectivity of the destination region set (i.e., X) with a limited number of RSUs allocated as the source nodes (i.e., S).

Theorem 1. For any $Q \in V(G)$ such that $s \in Q$, $\chi \notin Q$ and any S - χ -flow f , we have

- (a) $value(f) = \sum_{e \in \delta^+(Q)} f(e) - \sum_{e \in \delta^-(Q)} f(e)$,
 (b) $value(f) \leq \sum_{e \in \delta^-(Q)} u(e)$.

Proof. (a): Since the flow conservation rule holds for $v \in Q \setminus \{s\}$,

$$\begin{aligned} value(f) &= \sum_{e \in \delta^+(s)} f(e) - \sum_{e \in \delta^-(s)} f(e) \\ &= \sum_{v \in Q} \left(\sum_{e \in \delta^+(v)} f(e) - \sum_{e \in \delta^-(v)} f(e) \right) \\ &= \sum_{e \in \delta^+(Q)} f(e) - \sum_{e \in \delta^-(Q)} f(e); \end{aligned}$$

(b): this follows from (a) according $0 \leq f(e) \leq u(e)$ for every edge $e \in E(G)$. \square

Here, we can consider Q as a group of relay regions between the source RSUs and the destinations. The value of a maximum flow can not exceed the link capacity between the neighbor RSUs allocated.

4.3. RSU Allocation Strategy for Bidirectional Link Capacity

In this section, we will present our RSU allocation strategy design for the service capacity in the bidirectional traffic flow. The traditional Max-Flow-Min-Cut problem with single-source and single-destination can be solved with a polynomial-time algorithm. For example, the Edmonds–Karp algorithm, push–relabel algorithm, and relabel-to-front algorithm are proposed to calculate the maximal flow. The Store–Wagner algorithm is usually utilized to calculate the minimal cut.

As the urban traffic can be divided into multiple road segments, we can simplify the problem for each road segment, and analyze the relationship between the distributed RSUs and link capacity between them. Given a road segment with bidirectional traffic flow in Figure 3, we can construct an abstract ladder graph in Figure 4, including the bidirectional traffic flows of eastbound link and westbound link in each region. In Figure 4, the road segment can be divided into seven regions, as A, B, \dots , and G , respectively. The set of regions with eastbound traffic can be denoted as $V^e(G) = \{A^e, B^e, \dots, G^e\}$, and the set of regions with west bound traffic can be denoted as $V^w(G) = \{A^w, B^w, \dots, G^w\}$ accordingly. The link capacity (e.g., $|\widehat{AG}|$) is determined by the communication connectivity of passing-by regions. The communication connectivity in each region can be inferred from the communication time and vehicle density for V2V communication in the same direction, V2V communication in the opposite direction, and V2I communication. For node $v \in V(G)$, we denote $f^+(v)$ as the joining flow capacity, which presents the connectivity between the source node and node v . For example, we can have $f^+(C^e) = \min\{f^+(B^e), |\widehat{B^e C^e}|\} + \min\{f^+(C^w), |\widehat{C^w C^e}|\}$. In addition, we can denote $V^+(v)$ as the node regions before v along the traffic direction, and $V^-(v)$ as the node regions after v in the following traffic direction. For example, $V^+(D^e)$ denotes the node regions $\{A, B, C\}$, and $V^-(D^e)$ denotes the node regions $\{E, F, G\}$. In order to allocate RSUs $\{1, \dots, k\}$, we present the bidirectional allocation algorithm for RSUs in Algorithm 1.

As the connectivity bottleneck in the upstream will reduce the whole link capacity, the main idea in Algorithm 1 is to arrange the connectivity bottleneck in the downstream of the communication link through RSU allocation. The details of implementation issues will be illustrated through an example in Figure 4. Given a bidirectional traffic flow, the RSUs are allocated in region A and G . In the link \widehat{AG} , there exist four connectivity bottleneck nodes denoted as black nodes with $f^+(C^e) < f^+(F^w) < f^+(E^w) < f^+(D^e)$. We are motivated to maximize the link capacity $|\widehat{AG}|$ through RSU allocation. Initially, the set of regions B, C, D, E, F are available for RSU allocation. In step ①, $f^+(C^e)$ is selected as the link bottleneck, and the link is updated as \widehat{CG} ; in step ②, $f^+(F^w)$ is selected as the link bottleneck,

and the link is updated as \widehat{CF} ; in step ③, $f^+(E^w)$ is selected as the link bottleneck, and the link is updated as \widehat{CE} ; in step ④, $f^+(D^e)$ is selected as the link bottleneck, and the link is updated as \widehat{CD} . Finally, as $f^+(D^e) < f^+(C^w)$, the RSU is allocated in region D . As a result, \widehat{AG} can be divided into \widehat{AD} and \widehat{DG} with increasing link capacity $|D^wA^w|$ and $|D^eG^e|$, respectively.

Algorithm 1: Bidirectional Allocation Algorithm

```

1 for 1 to  $k$  do
2   Confirm  $\mathbb{S}_k$  as the set of available regions with minimal link capacity
3   while  $|\mathbb{S}| \neq 1$  do
4     Select  $f^+(v') = \min\{f^+(v) | v \in \mathbb{S}\}$  as the connectivity bottleneck
5     Update the link and  $\mathbb{S}_k = \mathbb{S}_k \cap V^-(v')$ 
6   end
7   Allocate the RSU  $k$  in region  $\mathbb{S}_k$ .
8   Delete the node from the set of available regions.
9   Divide the link into two parts with higher link capacity.
10 end
  
```

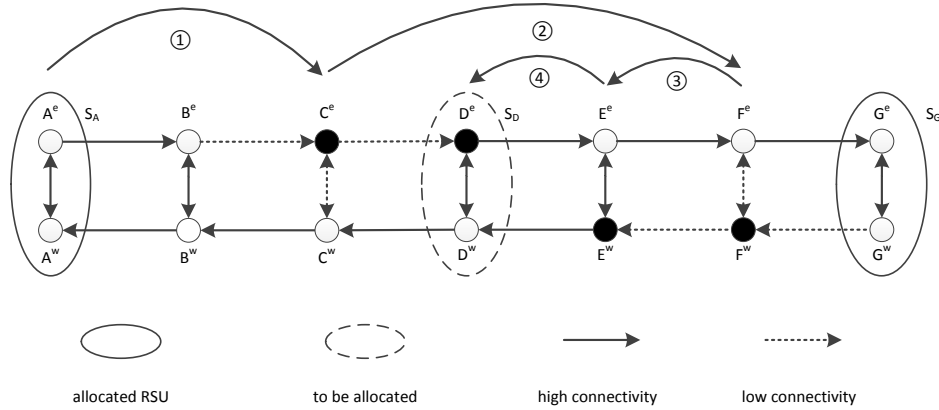


Figure 4. Ladder graph for bidirectional traffic flow.

5. Cache Replication

After the RSUs are allocated, the cache replication should be updated for the various request demands of the traffic flow. In this section, we will present the two-layer model and explore the replication strategy in a special case scenario and a general case scenario.

5.1. Two-Layer Mapping

We will present the system model and problem formulation for RSU content replication in VANETs. Without loss of generality, we assume that the set of RSUs with uniform capacity are distributed all over the urban areas. All the file items are of unit size, and each RSU can replicate items at most. Therefore, we have a mapping between the replicas and scattered RSUs, which can be denoted as a replication layer mapping. Furthermore, considering the content delivery, we can denote the joint time period for each moving vehicle to download its requested item passing by neighbor RSUs. We also assume that the connection between any vehicle and RSU is uniform in the coverage area, and the bandwidth capacity of RSUs is usually not considered as the constraint in VANETs according to previous research [16,22]. Therefore, the number of vehicles that can be served by a single RSU can be assumed to be unlimited, as these connections are transient for the moving traffic flow. Although there can be overlapped coverage areas between neighbor RSUs, we consider that the vehicles always connect to the nearest RSU. Generally, a vehicle needs to complete the content transmission through

several RSUs passing by that are replicated the same content item. Therefore, we can further have a transmission layer mapping between the RSUs and vehicles.

A two-layer mapping model is presented in Figure 5. The replication layer mapping indicates the content replication among different RSUs. The transmission layer mapping indicates the service connection between the vehicles and their passing-by RSUs. Given the two-layer mapping graph in Figure 5, we can consider the RSUs as the middle level connecting the replica nodes and the vehicle nodes. We can formulate the problem as a minimum replication vehicle cover problem, in which the request demand of vehicles should be supported with limited number of RSUs. This problem can be formulated in details as follows. We denote the set of replicas as $R = \{r_1, r_2, \dots, r_m\}$, the set of RSUs as $S = \{s_1, s_2, \dots, s_k\}$, and the set of vehicles as $V = \{v_1, v_2, \dots, v_n\}$. Our object is to find a replication layer mapping $f_1 = \{1, \dots, m\} \rightarrow \{1, \dots, k\}$ and a transmission layer mapping $f_2 = \{1, \dots, k\} \rightarrow \{1, \dots, n\}$ with $\sum_{i:f_1(i)=j} r_i \leq c$ for all $j \in \{1, \dots, k\}$ and $\sum_{j:f_2(j)=l} s_j \geq \sigma$ for all $l \in \{1, 2, \dots, n\}$ such that k is minimum. In the minimum replication vehicle cover problem, $\sum_{i:f_1(i)=j} r_i \leq c$ for all $j \in \{1, \dots, k\}$ represents the replication capacity limitation in the replication layer, and $\sum_{j:f_2(j)=l} s_j \geq \sigma$ guarantees the service time for all the vehicles in the transmission layer. This problem can be correlated to the conventional minimum weight vertex cover problem, which has been well known as the NP problem.

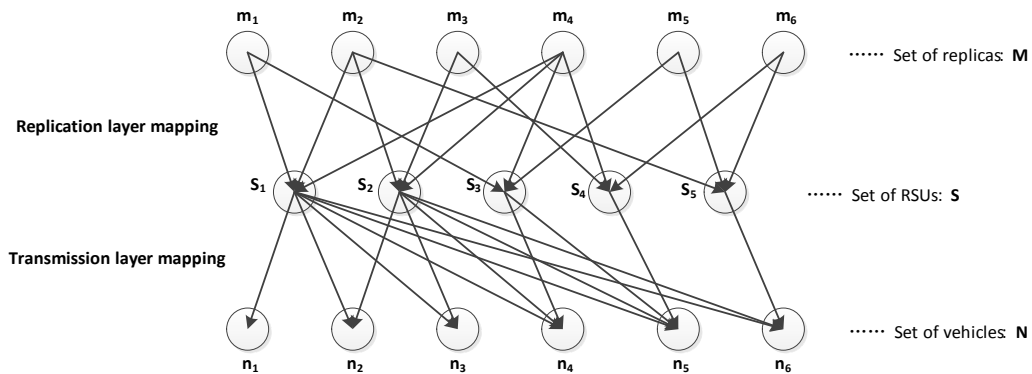


Figure 5. Two layer mapping model.

5.2. Online Implementation

In this section, we will present the online solution for practical implementations, which can be considered in a special case scenario and a general case scenario, respectively.

5.2.1. A Special Case Scenario

In the special case scenario, all of the vehicles can visit every RSU in sequence. In other words, we can redirect the request demand of a vehicle to an arbitrary RSU when it passes by the coverage area. Even though it is a special case, it also can be prevalent in the real world, especially the highway, in which the vehicles can drive a long distance and RSUs can be allocated along the road. Therefore, if we do not consider the delay constraint, in this special case scenario, our problem can be transmitted into a minimum weight set cover problem, in which the storage capacity can be considered as the weight of each RSU. As we assume that all the RSUs are with equal storage capacity like previous works [13,14], the weight of the set can refer to the number of RSUs in our formulated problem. Accordingly, we can present an adaptive greedy replication (AGR) for a special case scenario in Figure 6.

As we assume that the traffic flow would pass all the RSUs in a sequence along the road, the popularity distribution of replication content in cloud servers can be recorded in advance. The popularity list can be achieved for each replica, and the replicas can be migrated among the RSUs in a sequence. In the adaptive greedy replication strategy, the replica with the highest popularity in the current situation would be allocated in the RSU until the storage capacity is full. After each

round of replication, the corresponding request demand of the vehicles can be served by this RSU when they pass by. This procedure is implemented at each RSU in a sequence until all of the request demands of the traffic flow are satisfied. Therefore, the replicas with high popularity would be allocated at the head of the RSU sequence, and the ones with low popularity are allocated at the end of the RSU sequence. During practical implementation, the replicas in the sequence of RSUs can be migrated adaptively to adjust with time-varying request demands. According to the request demands, the adaptive greedy replication strategy can start a new RSU or terminate an old RSU at the tail of the sequence. In other words, our proposed greedy algorithm is straightforward enough to allocate the most popular replicas in priority, and adaptively redirect the request demands towards these RSUs to maximize the service time.

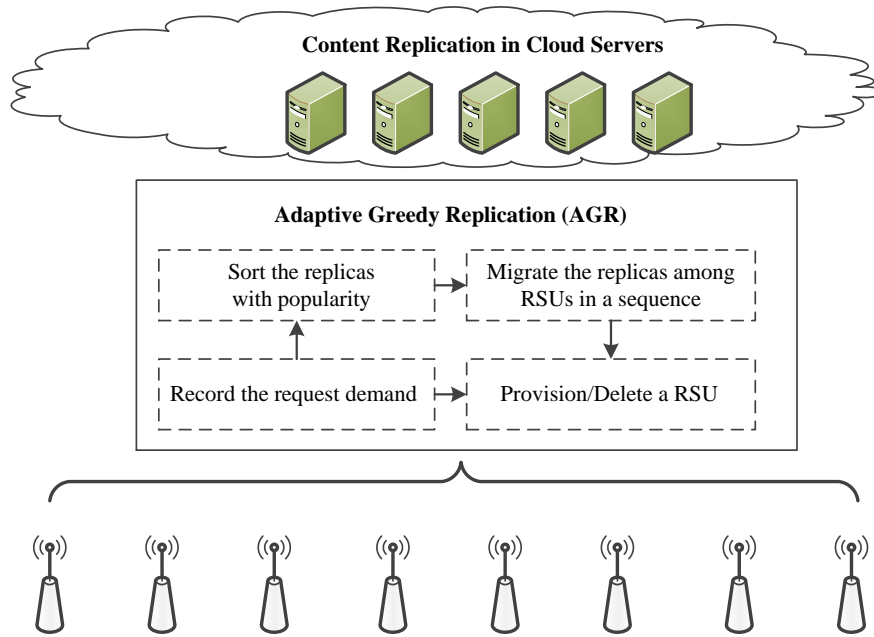


Figure 6. Adaptive greedy replication.

Theorem 2. For any instance (R, S, V) , the greedy algorithm finds a set cover with a result at most $H(\lambda) \cdot \text{OPT}(R, S, V)$, where $\lambda = \max_{s_k \in S} |s_k|$ and $H(\lambda) = 1 + 1/2 + \dots + 1/\lambda$.

Proof. Let (R, S, V) be an instance with $\{s_1, s_2, \dots, s_k\}$ as the solution found by the adaptive greedy solution, where η_i is the set chosen in the i -th iteration. For $j = 0, \dots, k$, let $W_j = \bigcup_{i=1}^j \eta_i$. For each replica e , let $j(e) = \min\{j \in \{1, \dots, k\} | e \in R_j\}$ be the iteration where replica e is covered, and $y(e)$ be the popularity. Given an RSU allocation S and $k' = \max\{j(e) | e \in S\}$, we can have

$$\begin{aligned} \sum_{e \in S} y(e) &= \sum_{i=1}^{k'} \sum_{j(e)=i} y(e) \leq c(S) \cdot \sum_{i=1}^{k'} \frac{\varepsilon_i - \varepsilon_{i+1}}{\varepsilon_i} \\ &= c(S) \cdot \sum_{i=1}^{k'} \left(\frac{1}{\varepsilon_i} + \frac{1}{\varepsilon_i - 1} + \dots + \frac{1}{\varepsilon_{i+1}} \right) \leq c(S) \cdot H(\varepsilon_1), \end{aligned}$$

where $\varepsilon_i = |S \cap W_{i-1}|$. As $\varepsilon_i = |S| \leq \lambda$, we can have $\sum_{e \in S} y(e) \leq c(S) \cdot H(\lambda)$. Furthermore, we sum over all $S \in \text{OPT}$ for an optimum set cover OPT and have

$$c(\text{OPT}) \cdot H(\lambda) \geq \sum_{S \in \text{OPT}} \sum_{e \in S} y(e) \geq \sum_{e \in U} y(e).$$

□

We can see that the greedy algorithm can achieve an optimal solution if $y(e)$ is constant for every $e \in U$ (i.e., the popularity is equal for every replica). In addition, the adaptive greedy strategy is efficient especially when there are a great number of replicas or RSUs to be matched.

However, there are also some limitations for the proposed adaptive greedy algorithm to be implemented in the large scale urban areas. First, we have to know the popularity of each replica to maintain the popularity list. Second, the proposed adaptive greedy strategy is dependent on the assumption that all the vehicles can request the RSUs in a sequence along the road. Actually, the traffic-in and traffic-out in the road will both change the mapping relationship between the RSUs and vehicles in the transmission layer. Third, the centralized solution would bring extra overhead to frequently update the popularity among different coverage areas. Above all, we need to further consider the general case scenario especially under the dynamic traffic flow in the next section.

5.2.2. A General Case Scenario

In order to deal with a general case scenario with dynamic traffic flow in global urban areas, we present a distributed online scheduling strategy for a general case scenario in Figure 7, in which the vehicle may enter or leave the road at anytime and anywhere as a traffic-in or traffic-out. Each RSU can monitor the traffic flow and request demand in its local coverage areas, and the statistic information is only shared among neighbor RSUs for cooperation. Accordingly, the distributed online scheduling strategy includes a request scheduling mechanism for the transmission layer mapping, and a replication scheduling mechanism for the replication layer mapping, respectively.

Replication Layer Mapping. The replication scheduling mechanism refers to determining the mapping relationship between the RSU and replication content. Given a road segment, the RSUs are allocated in sequence along the road, and the vehicles can pass by the coverage areas of these RSUs in order. Thus, for every RSU at any time, the traffic flow under its coverage areas consists of either the vehicles from previous RSUs, or the newly joint vehicles in this distance. Without the global popularity of each replica, the replication scheduling mechanism can be designed according to the local popularity prediction, which is influenced by the statistics of vehicles from previous neighbor RSUs and newly joint vehicles. We can have a comprehensive consideration of the replication scheduling with the estimation of the two factors as follows:

$$P_k(t+1) = \sigma_{k-1}(t) \cdot P_{k-1}(t) + P'_{k-1}(t),$$

where $P_k(t+1)$ and $P_{k-1}(t)$ represent popularity distribution of RSU k at time $t+1$ and popularity distribution of RSU $k-1$ at time t , respectively. Furthermore, we have $P'_{k-1}(t)$ as the popularity distribution of newly joint vehicles in the distance between the RSU $k-1$ and RSU k . In addition, we denote $0 \leq \sigma_{k-1}(t) \leq 1$ as the popularity distribution influence of the previous neighbor RSU. Larger $\sigma_{k-1}(t)$ indicates a lesser fraction of traffic flow, which leaves the road after passing by previous RSUs. Generally, we consider $\sigma_{k-1}(t)$ as a time-varying coefficient according to the dynamic traffic flow.

Transmission Layer Mapping. The request scheduling mechanism refers to determining the mapping relationship between the RSUs and moving vehicles. Ideally, each vehicle can request its demand from the passing-by RSUs with requested content replication. Nevertheless, for each RSU, there usually exists a connection limitation to serve the request demands in its coverage area at the same time. Then, the number of connections can be a bottleneck for the transmission between the RSUs and vehicles. Generally, we utilize the request scheduling to arrange these subsequent request demands as follows: (1) New Requester First: the RSUs will serve the newly-joint vehicles in priority; (2) Old Requester First: the RSUs will serve the vehicles who drive a long distance in priority; (3) First-in First-out: the RSUs will serve the requester in a regular order.

Whenever a vehicle joins the traffic flow of a road, it will send a request demand to the coherent RSU. After receiving these requests, the RSU will update its local replication and complete the request scheduling. Accordingly, we can consider the practical issues of online implementation from the

vehicle side and RSU side, respectively. From the viewpoint of vehicles, the passing-by RSU will be recorded in a sequence of labels. For example, $\langle r1, s1, t1 \rangle$ and $\langle r1, s2, t2 \rangle$ represent the fact that this vehicle sends a request to RSU $s1$ and $s2$ at time $t1$ and $t2$ for replica $r1$. From the viewpoint of RSUs, the labels will be received online with the request demand, and the statistics of the dynamic traffic flow can be reflected from these labels. Correspondingly, the replication scheduling and request scheduling will be implemented at the RSUs in a distributed manner.

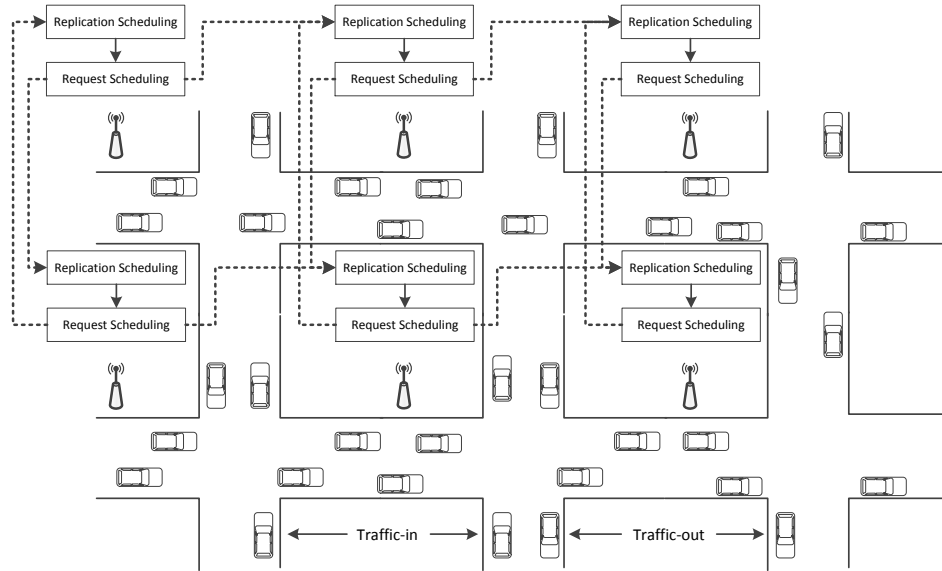


Figure 7. Distributed online replication.

6. Performance Evaluation

Our proposed adaptive RSU scheduling solution will be evaluated with other RSU distribution strategies through numerical results. We will present the simulation environment with time-varying bidirectional traffic flows, as well as various request demands. Generally, we have two types of simulation environments as a region-level simulation for an urban area, and a vehicle-level simulation for a road segment.

6.1. Region-Level Simulation Environment

In the region-level simulation environment, we consider an urban area that can be divided into 100 different regions. In each region, we assume that the bidirectional traffic flows are independent from each other, and follow a poisson distribution with λ^e for the eastbound traffic flow and λ^w for the westbound traffic flow. Thus, we can have a group of parameters as $\{\lambda_1^e, \lambda_2^e, \dots, \lambda_{100}^e\}$ for the eastbound traffic flow and $\{\lambda_1^w, \lambda_2^w, \dots, \lambda_{100}^w\}$ for the westbound traffic flow, which demonstrate the vehicle density for each area. Furthermore, we assume that the parameters $\{\lambda_1, \lambda_2, \dots, \lambda_{100}\}$ follow a normal distribution with expectation μ and variance σ . As the variance σ increases, the vehicle density tend to be non-uniformly distributed among different areas. By default, we can set the coverage ratio in each area as 0.005 for V2V communications and 0.5 for V2I communications. The details of the parameter configuration are presented in Table 1.

Furthermore, we assume 1000 different content replicas with Zipf distribution to simulate the various request demands. The popularity (or request rate) of the i -th replica is $\eta_i = i^{-s} / \sum_{k=1}^M k^{-s}$. The value of parameter s is usually in the range of $0.271 \leq s \leq 1$, according to previous application services. Here, we can consider different skewness to simulate popularity distribution of replicas. We present the popularity distribution of the top-30 replicas with different parameter s setup in Figure 8. Without loss of generality, each RSU has a uniform storage capacity for content replication.

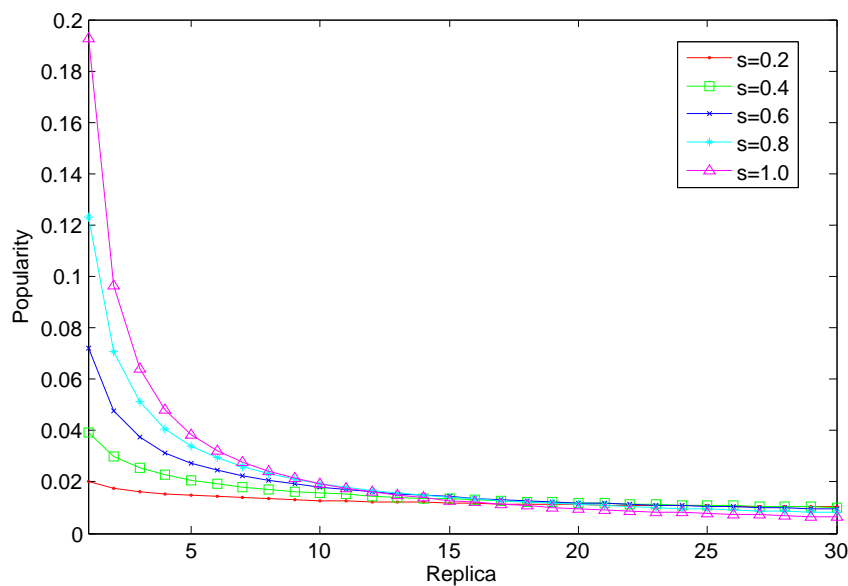


Figure 8. Skewness with different parameter s .

Table 1. Region-level parameter configuration.

Parameter	Configuration
Number of regions	100
Area of each region	$1000 \times 1000 \text{ m}^2$
Number of vehicles expectation in each region	$100 \leq \mu \leq 300$
Number of vehicles variance in each region	$5 \leq \sigma \leq 30$
Communication range	400 m
Number of replicas	1000
Skewness range	$0.271 \leq s \leq 1$

6.2. RSU Allocation

In this section, we will evaluate our proposed solution with other RSU allocation strategies in the simulation environment. We consider the Top-K strategy (i.e., the RSUs are allocated in the most density areas.) and uniform distribution strategy (i.e., the RSUs are allocated along the road segment with uniform distance.) as the benchmark strategies for comparison.

6.2.1. Impact of RSU Numbers

To analyze the impact of RSU distribution, we evaluate these strategies with RSU numbers $m = 10$, $m = 20$, and $m = 40$, respectively. As to the vehicle density, we assume that the number of vehicles in each region $\{\lambda_1, \lambda_2, \dots, \lambda_{100}\}$ follows the normal distribution with $\sigma = 10$ and expectation μ from 100 to 300. The relative connectivity is used as the performance metric to compare the three strategies. In Figure 9a, we can see that the difference among these strategies is minor with a few number of RSUs. As the number of RSUs grows, the performance gap is increased. In Figure 9c, the number of RSU is $m = 40$, and the performance our proposed bidirectional allocation strategy is much better than the other two strategies, especially with the low vehicle density. Generally, the Top-K strategy has a better performance compared with the uniform distribution strategy, as the Top-K strategy always allocates the RSUs in the regions with the highest vehicle density. Nevertheless, the performance difference is reduced as the number of RSUs keeps growing. We can conjecture that there would be more opportunity for vehicle-to-vehicle communication when the region has a higher vehicle density. Therefore, our Top-K strategy can hardly improve the global connectivity through naively allocating the RSUs in the regions with highest vehicle density.

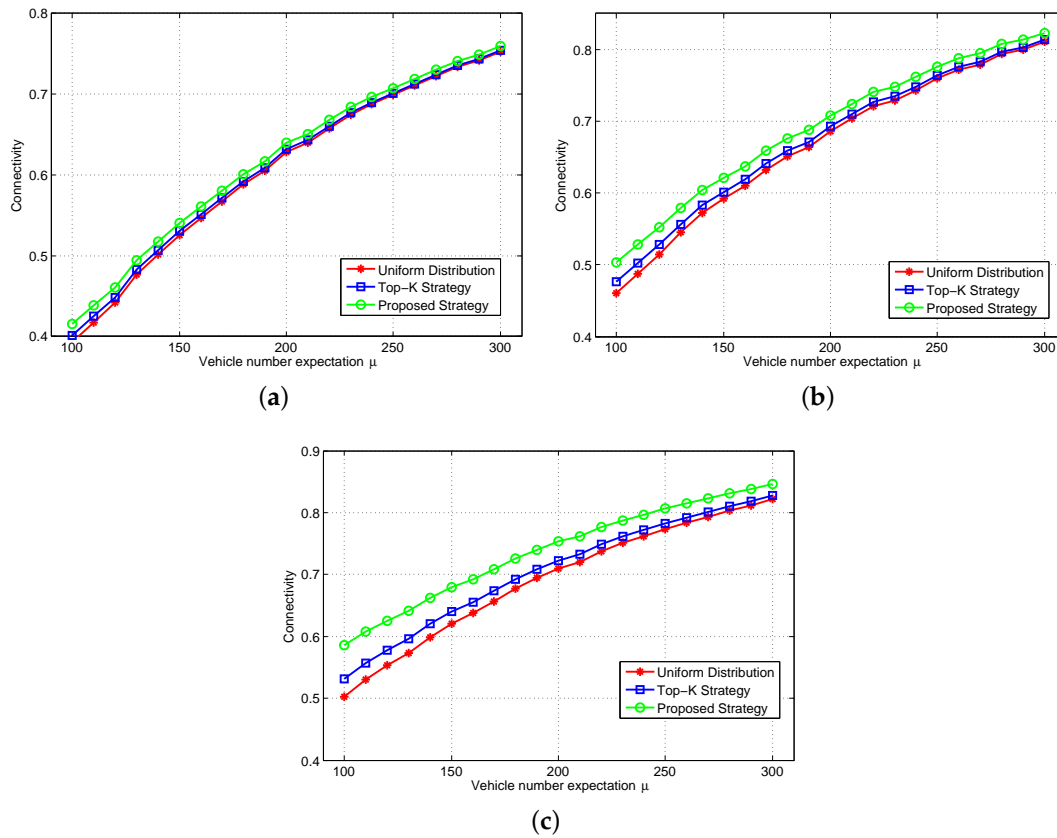


Figure 9. Connectivity under different allocation strategies. (a) number of RSUs $m = 10$; (b) number of RSUs $m = 20$; (c) number of RSUs $m = 40$.

6.2.2. Impact of Traffic Condition

To analyze the impact of vehicle distribution, we will compare the connectivity of these strategies under the vehicle distribution variation. Here, we have the number of vehicle expectation $\mu = 100$ with variance σ from 0 to 30 in Figure 10. When $\sigma = 0$, the traffic flow is uniformly distributed among all the regions. In this situation, the performance gap among RSU allocation strategies is minor. As the variance increases to $\sigma = 10$ and $\sigma = 15$, we can see that both our proposed strategy and Top-K strategy have greatly improved connectivity, and our proposed strategy obviously has better performance. Meanwhile, the performance of the uniform distribution strategy deteriorates as the variation increases. Furthermore, as the variance continues to increase at $\sigma = 20$ and $\sigma = 30$, the connectivity begins to converge in this period, and the performance gain between our proposed strategy and Top-K strategy is minor. We can conjecture that when the traffic density is low and uniform, both our proposed bidirectional allocation strategy and Top-K strategy would make similar decisions to allocate the RSUs in the region with higher vehicle density.

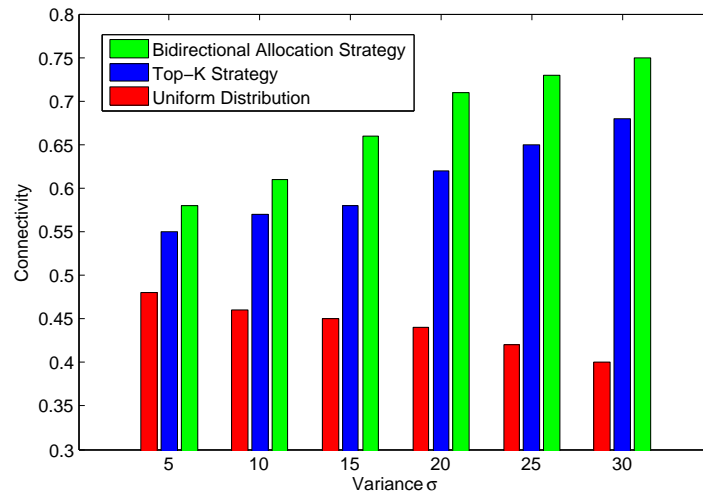


Figure 10. Connectivity with vehicle density variation.

6.3. Cache Replication

In this section, we will evaluate our proposed solution with other cache replication strategies in the simulation environment. We consider the Top-K strategy (i.e., the RSUs are replicated with the most popular k replicas.), and the uniform distribution strategy (i.e., the replicas are replicated with equality possibility.) for comparison.

6.3.1. Special Case Scenario

In this section, we will evaluate the system performance in a special case scenario, in which the popularity distribution of content replica can be known in advance, and the request demand can be redirected to any RSU if possible. Accordingly, we can have the number of minimal RSUs deployed in the optimal solution as the benchmark, and compute the relevance of other three strategies in Figure 11. When $s = 0.2$, we can have a uniform distribution with an almost equal popularity for each replica. In this case, the proposed adaptive greedy strategy and uniform distribution can have an approximate optimal solution. This is because each replica may have similar request demands. Meanwhile, the Top-K solution has to deploy more RSUs for the coverage of the whole service, as only the most popular content is replicated and the popularity difference is minor. As the skewness becomes higher, we can see that the Top-K strategy will have a better performance and the uniform distribution strategy will have a worse performance. On the other hand, our proposed adaptive-greedy strategy can have a close-to-optimal performance, and can be scalable to the change of request popularity skewness.

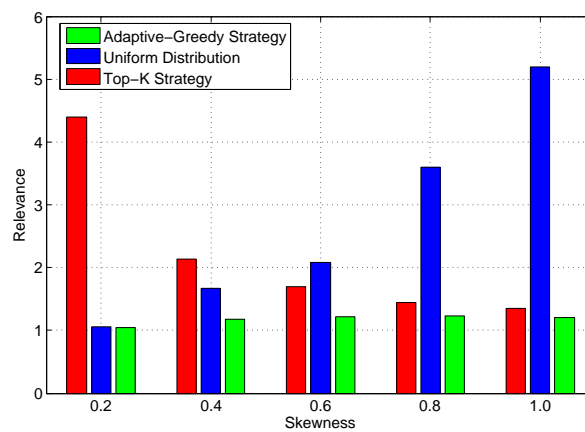


Figure 11. Performance comparison with different skewness.

6.3.2. General Case Scenario

In this section, we will evaluate the system performance for a general case scenario, in which the vehicles can join or leave the system anytime and anywhere, and the request demand can only be redirected to the passing-by RSUs under the connection capacity limitations. We consider that the number of vehicles to join or leave between every pair of RSUs follows a Poisson distribution at each time slot. The online strategy including replication scheduling and request scheduling is implemented to deal with the time-varying traffic flow. Specifically, we consider three types of mechanisms (i.e., First-in First-out, Old Requester First, and New Requester First, respectively) in request scheduling. We compare the cumulative distribution function (CDF) of a completed request with different numbers of passing-by RSUs under these mechanisms in Figure 12. Generally, all three of these mechanisms would have an increasing CDF as the number of passing-by RSUs grows. Compared with First-in First-out and New Requester First mechanism, Old Requester First has a fast CDF increase with a limited number of RSUs, as the vehicles passing-by previous RSUs would be served in priority. Nevertheless, the CDF increase of Old Requester First becomes slow as the number of passing-by RSUs keeps growing. On the contrary, New Requester First would have a slow CDF increase initially, and a fast increase can be observed with an increasing number of RSUs. As New Requester First serves the newly joint vehicles in priority, it is suitable for more RSUs allocated. During the real-world implementation, these request scheduling strategies can be flexibly adjusted according to the RSU distribution and traffic condition.

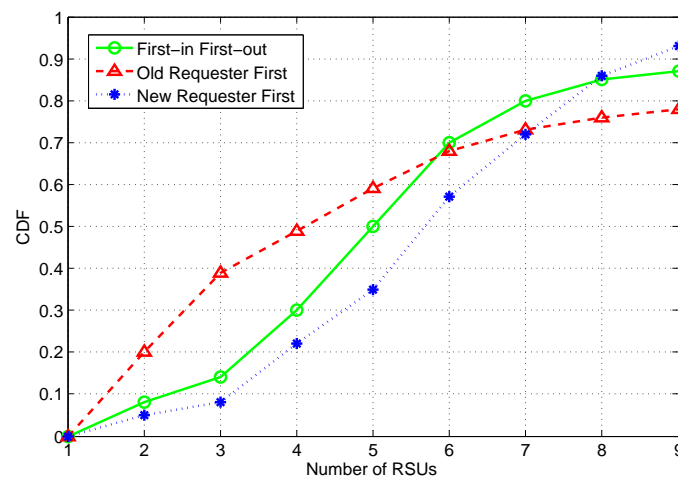


Figure 12. Cumulative distribution functions with different RSUs.

6.4. Vehicle-Level Online Implementation

In this section, we will present the vehicle-level simulation to evaluate our proposed adaptive bidirectional scheduling strategy (ABS for short). Given a specific road segment, we assume that the number of vehicles entering the road segment follows a Poisson distribution with parameter λ . When the vehicles are driving along the road, they can have V2I communications with RSUs or V2V communications with other vehicles in 6 Mbps bandwidth and 400 m communication range. We also assume that the total number of replicas is 10,000, and the file size of each replica is 200 m on average. The storage capacity of each RSU is 10T, which means that a single RSU can replicate 500 different replicas approximately. In addition, the storage capacity of each vehicle is 1T, which means a single vehicle can replicate five different replicas in a carry-and-forward manner. The experiment configuration is listed in Table 2. To evaluate the performance, we use a cellular based strategy as the benchmark, in which the vehicles will resort to the long-range communications through a cellular network. In addition, we have some other strategies for comparison. PROPHET is a prediction based

solution for intermittently connected networks [28]. The V2X based approach is a comprehensive solution with V2V and V2I communications to reduce the time delay [29]. DBC is a density-based multilevel clustering algorithm to formulate stable and long living clusters for VANETs [30]. CMCS is a connectivity-oriented maximum coverage RSU deployment scheme, which aims to maximize V2I communication performance in urban areas [14]. HE is a heuristic solution with greedy algorithm for the highway scenario [15].

Table 2. Vehicle-level parameter configuration.

Parameter	Value
Length of road segment	50 km
Vehicle velocity	0–100 km/h
Communication range	400 m
Average bit rate	6 Mbps
Number of replicas	10,000
Average size of each replica	200 M
Average storage capacity of each vehicle	2 G
Average storage capacity of each RSU	10 T
Simulation time	2 h

In the vehicle-level simulation environment, we compare the cost saving and time delay in Table 3. Specifically, the cost savings refer to the content delivery through V2I or V2V rather than cellular communications. These strategies are implemented in a two-hour simulation for the performance on average and worst case. PROPHET and DBC are implemented with multi-hop data dissemination in VANETs environment. Without the support of RSUs, we can see that PROPHET and DBC have a lower performance on cost savings and time delay than RSU-based solutions. Specifically, our proposed strategy has higher cost savings than V2X, while V2X can reduce time delay more efficiently. We can conjecture the reason that our proposed ABS is designed to improve the global RSU resources utilization efficiency, and V2X is designed to reduce time delay with the combination of V2I and V2V communications. In addition, CMCS and HE can have 60.3% and 55.2% with RSU deployment. Both of them assume the highway scenario in which all the vehicle traffic follows in the same direction, without considering the V2V communications in bidirectional traffic flows. Obviously, our proposed solution with highest cost savings is more V2V friendly, in spite of a long time delay to balance the content allocation priority.

Table 3. Relative cost and time delay comparison.

Scheduling Strategy	Cost Saving		Time Delay	
	Average	Worst Case	Average	Worst Case
PROPHET	43.2%	23.2%	10.8 min	23.3 min
DBC	44.4%	17.6%	8.7 min	25.5 min
V2X	62.8%	11.2%	5.4 min	12.5 min
ABS	73.7 %	6.3 %	16.8 min	42.2 min
CMCS	60.3%	14.3 %	11.4 min	30.9 min
HE	55.2%	13.1 %	12.8 min	23.7 min

7. Conclusions

With the prevalence of autonomous vehicle technology, the distributed RSUs can play a more significant role in the modern city environment. They are not only responsible for message broadcasts, but also can facilitate data dissemination for the large file size content. The efficiency of data dissemination in VANETs' environment will be influenced by the content replication among these distributed infrastructure resources.

In this paper, we considered a cloud-based architecture, in which the virtual RSUs can act as the front edge nodes of the cloud platform. The RSU-level cooperation was analyzed in the dynamic environment including both the traffic conditions and request demand distribution. With the exploration of bidirectional vehicle traffic flow, a bidirectional allocation algorithm was presented to allocate the RSUs among different regions with time-varying link capacity. In addition, we formulated content replication in distributed RSUs as the minimum replication set coverage problem in a two-layer mapping model, and analyzed the solutions in different scenarios. Numerical results further proved the superiority of our proposed solution, as well as the scalability to traffic condition variations.

In order to analyze the service efficiency under diverse traffic conditions, we assume that each type of data communication corresponds to the vehicle velocities in the same or opposite directions, and the link connectivity in each region refers to the data communications and local vehicle density. This simplified model can provide a straightforward view to analyze the impact of traffic conditions in an urban area. On the other hand, it sacrifices the details of the interrupt connections in VANETs. In our future work, we will further explore the RSU allocation and content replication under the impact of the characters in the wireless communications (i.e., the long-range V2I communications and the short-range V2V communications).

Acknowledgments: This research is supported by the following grants: National Natural Science Foundation of China (Grant No. 61602214, 61672264, 61702227), Natural Science Foundation of Jiangsu Province in China (Grant No. BK20160191), National Key Research and Development Program of China (Grant No. 2016YFB0800801), National Science and Technology Support Program of China (Grant No. 2015BAH54F01), Research Innovation Program for College Graduates of Jiangsu Province (Grant No. SJZZ16_0218, KYLX16_769), Research and Practice on Graduate Educational Reform of Jiangsu Province (Grant No. JGZZ16_065).

Author Contributions: F. Chen and Z. Hua contributed the algorithm design; F. Chen and Y. Liu conceived and designed the experiments; X. Bi performed the experiments; R. Lyu and X. Zhang analyzed the data; F. Chen wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zheng, K.; Zheng, Q.; Chatzimisios, P.; Xiang, W.; Zhou, Y. Heterogeneous Vehicular Networking: A Survey on Architecture, Challenges, and Solutions. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2377–2396.
2. Vegni, A.M.; Loscri, V. A Survey on Vehicular Social Networks. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2397–2419.
3. Tao, J.; Zhang, Z.; Feng, F.; He, J.; Xu, Y. Non-Cooperative Resource Allocation Scheme for Data Access in VANET Cloud Environment. In Proceedings of the International Conference on Advanced Cloud and Big Data, Yangzhou, China, 30 October–1 November 2015.
4. Rehman, O.; Ould-Khaoua, M.; Bourdouce, H. An Adaptive Relay Nodes Selection Scheme for Multi-Hop Broadcast in VANETs. *Comput. Commun.* **2016**, *87*, 76–90.
5. Qureshi, F.F.; Iqbal, R.; Asghar, M.N. Energy efficient wireless communication technique based on Cognitive Radio for Internet of Things. *Elsevier J. Netw. Comput. Appl.* **2017**, *89*, 14–25.
6. Doolan, R.; Muntean, G.M. EcoTrec—A Novel VANET-Based Approach to Reducing Vehicle Emissions. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 608–620.
7. Lee, E.; Lee, E.-K.; Gerla, M.; Oh, S.Y. Vehicular Cloud Networking: Architecture and Design Principles. *IEEE Commun. Mag.* **2014**, *52*, 148–155.
8. Zhang, D.; Chow, C.Y.; Li, Q.; Zhang, X.; Xu, Y. SMashQ: Spatial Mashup Framework for k -NN Queries in Time-Dependent Road Networks. *Distrib. Parallel Databases* **2013**, *31*, 259–287.
9. Said, A.M.; Marot, M.; Ibrahim, A.W.; Afifi, H. Modeling Interactive Real-time Applications in VANETs with Performance Evaluation. *Elsevier Comput. Netw.* **2016**, *104*, 66–78.
10. Benrhaïem, W.; Hafid, A.S.; Sahu, P.K. Multi-hop Reliability for Broadcast-Based VANET in City Environments. In Proceedings of the IEEE International Conference on Communications (ICC), Kuala Lumpur, Malaysia, 22–27 May 2016.
11. Zhang, D.; Chow, C.Y.; Li, Q.; Zhang, X.; Xu, Y. A Spatial Mashup Service for Efficient Evaluation of Concurrent k -NN Queries. *IEEE Trans. Comput.* **2016**, *65*, 2428–2442.

12. Bitam, S.; Mellouk, A.; Zeadally, S. VANET-cloud: A generic cloud computing model for vehicular Ad Hoc networks. *IEEE Wirel. Commun.* **2015**, *22*, 96–102.
13. Wu, T.J.; Liao, W.; Chang, C.J. A Cost-Effective Strategy for Road-Side Unit Placement in Vehicular Networks. *IEEE Trans. Commun.* **2012**, *60*, 2295–2303.
14. Ou, D.; Yang, Y.; Xue, L.; Dong, D. Optimal Connectivity-Based Deployment of Road Side Units for Vehicular Networks in Urban Environments. *Transp. Res. Rec. J. Transp. Res. Board* **2016**, *2559*, 46–56.
15. Balouchzahi, N.M.; Fathy, M.; Akbari, A. Optimal road side units placement model based on binary integer programming for efficient traffic information advertisement and discovery in vehicular environment. *IEEE IET Intell. Transp. Syst.* **2015**, *9*, 851–861.
16. Abdrabou, A.; Zhuang, W. Probabilistic Delay Control and Road Side Unit Placement for Vehicular Ad Hoc Networks with Disrupted Connectivity. *IEEE J. Sel. Areas Commun.* **2011**, *29*, 129–139.
17. Liu, Y.; Niu, J.; Ma, J.; Shu, L.; Hara, T.; Wang, W. The insights of message delivery in VANETs with a bidirectional traffic model. *Elsevier J. Netw. Comput. Appl.* **2013**, *36*, 1287–1294.
18. Wang, J.; Liu, K.; Sha, E.H.M. An Efficient Cluster-Based Data Sharing Algorithm for Bidirectional Road Scenario in Vehicular Ad-Hoc Networks. In Proceedings of the International Conference on Algorithms and Architectures for Parallel Processing, Zhangjiajie, China, 18–20 November 2015.
19. Akand, M.M.R.; Nayeem, M.T.; Sumon, M.R.; Alam, M.M. A probabilistic delay model for bidirectional vanets in city environments. In Proceedings of the IEEE Conference on Vehicular Technology Society Asia Pacific Wireless Communications Symposium, Seoul, Korea, 22–23 August 2013.
20. Lee, J.; Jeong, J.; Du, D.H.C. Link Delay Modeling for Two-Way Traffic Road Segment in Vehicular Networks. In Proceedings of the International Conference on Internet of Vehicles, Chengdu, China, 19–21 December 2015.
21. Lim, S.; Chae, S.H.; Yu, C.; Das, C.R. On Cache Invalidation for Internet-based Vehicular Ad Hoc Networks. In Proceedings of the IEEE International Conference on Mobile AdHoc and Sensor Systems, Atlanta, GA, USA, 29 September–2 October 2008.
22. Ding, R.; Wang, T.; Song, L.; Han, Z.; Wu, J. Roadside-Unit Caching in Vehicular Ad Hoc Networks for Efficient Popular Content Delivery. In Proceedings of the IEEE Wireless Communications and Networking Conference, New Orleans, LA, USA, 9–12 March 2015.
23. Wang, M.; Liang, H.; Deng, R.; Zhang, R.; Shen, X. VANET Based Online Charging Strategy for Electric Vehicles. In Proceedings of the IEEE Global Communications Conference (GLOBECOM), Atlanta, GA, USA, 9–13 December 2013.
24. Mukherjee, J.C.; Gupta, A.; Sreenivas, R.C. Event Notification in VANET with Capacitated Roadside Units. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 1867–1879.
25. Azizian, M.; Cherkaoui, S.; Hafid, A. An Optimized Flow Allocation in Vehicular Cloud. *IEEE Access* **2016**, *4*, 6766–6779.
26. Feteiha, M.F.; Hassanein, H.S. Enabling Cooperative Relaying VANET Clouds Over LTE-A Networks. *IEEE Trans. Veh. Technol.* **2015**, *64*, 1468–1479.
27. Kim, R.; Lim, H.; Krishnamachari, B. Prefetching-Based Data Dissemination in Vehicular Cloud Systems. *IEEE Trans. Veh. Technol.* **2016**, *65*, 292–306.
28. Lindgren, A.; Doria, A.; Schelen, O. Probabilistic routing in intermittently connected networks. *ACM SIGMOB. Mob. Comput. Commun. Rev.* **2003**, *7*, 19–20.
29. Mostafa, A.; Vegni, A.M.; Singoria, R.; Oliveira, T.; Little, T.D.C.; Agrawal, D.P. A V2X-based approach for reduction of delay propagation in vehicular Ad-Hoc networks. In Proceedings of the 11th International Conference on ITS Telecommunications, St. Petersburg, Russia, 23–25 August 2011.
30. Kuklinski, S.; Wolny, G. Density Based Clustering algorithm for Vehicular Ad-Hoc Networks. *Int. J. Internet Protoc. Technol.* **2009**, *4*, 149–157.

