Keeping Data Alive: Communication Across Vehicular Micro Clouds

Gurjashan Singh Pannu*, Florian Hagenauer*, Takamasa Higuchi†, Onur Altintas† and Falko Dressler*
*Heinz Nixdorf Institute and Dept. of Computer Science, Paderborn University, Germany
†Toyota InfoTechnology Center U.S.A., Mountain View, CA
{pannu,hagenauer,dressler}@ccs-labs.org, {ta-higuchi,onur}@us.toyota-itc.com

***Abstract*—Vehicular micro clouds are considered a prime building block for next generation Intelligent Transportation Systems (ITS) also supporting a variety of Information and Communication Systems (ICT) applications in smart cities. Such micro clouds are established by multiple cars equipped with communication, storage, and computational resources. We recently presented the concept of hierarchical vehicular cloud computing, which is meant to extend on Mobile Edge Computing (MEC). Based on clustering algorithms, we can set up and maintain such micro clouds and eventually make use of the distributed resources. Looking at the high mobility of cars, it is very difficult to reliably maintain data collected by cars at a given location in space that is geographically relevant, e.g., at intersections. In this paper, we propose a new protocol which encourages coordination between neighboring micro clouds to help keeping local data current, i.e., cars moving out of the micro cloud may take data to neighboring clouds, hand data over to cars moving towards the original micro cloud, and, thus, returning the data to its original geographical location. We evaluate the performance of the protocol with different vehicle densities in a Manhattan Grid scenario and our results show the benefits of our proposed inter micro cloud coordination protocol.**

***Index Terms*—Mobile Edge Computing, Vehicular Cloud, Vehicular Micro Cloud, Data Management**

I. Introduction

There has been a recent shift in focus of vehicular networking research community towards applications supporting cooperative driving [1] and cooperative perception [2]. This is further supported as modern cars are equipped with wide range of sensors, computing, networking, and storage resources. The sensing on-board units sense abundant data from the surrounding, which can also be cooperatively used by nearby vehicles, bicyclists, and even pedestrians. Cooperative sensing can be used to maintain live 3D maps and even make complex maneuvers safely. The powerful configuration of cars makes them an important Information and Communication Systems (ICT) resource, transforming the Intelligent Transportation Systems (ITS) in future smart cities.

The generated data needs to be stored and also requires frequent updates. Uploading the data to data centers is beneficial for further analysis using advanced data analytic techniques. However, vehicles interested in the data also experience longer end-to-end delays when downloading directly from data centers.

To solve similar problems in cellular mobile networks, the Mobile Edge Computing (MEC) [3] architecture has been proposed. The underlying idea is to provide computing and storage capabilities at the edge of the cellular network, which is in close proximity to the users. In vehicular networks, Eltoweissy et al. [4], Gerla [5], and Dressler et al. [6] proposed the concept of vehicular cloud computing, which later evolved as the vehicular micro cloud architecture [7], [8]. In simple words, cars cooperatively form a small cluster called *vehicular micro cloud* which offers computing and storage services to nearby cars, pedestrians, and bicyclists, thereby, extending the concept of MEC in vehicular networks. Vehicular micro clouds can be mobile (formed by cars moving in same direction) or stationary (formed at a certain geographic region).

The data stored in geographically stationary vehicular micro clouds is usually relevant to a certain geographic location. It can be uniquely identified, thus favoring communication models like Information Centric Networking (ICN) [9] and Named Data Networking (NDN) [10] rather than typical host-centric communication models.

One of the main open research questions in vehicular micro cloud research is to keep the data belonging to a certain micro cloud available in the micro cloud. Each data content is associated with a unique vehicular micro cloud, which we call *parent micro cloud*. Our goal is to keep the data within the parent micro cloud as long as possible. This becomes very challenging due to vehicular mobility. Cars join the stationary micro cloud, collect some data, and after a while, they leave. If the car leaving the micro cloud is the last one to have certain data, then the data becomes non-recoverable in the parent micro cloud. The variation in traffic density adds more challenges on top of it. The micro cloud region can get over-crowded by cars at certain times, and within a fraction of a minute, all cars can leave the micro cloud too.

In this paper, we fill in missing gaps in vehicular micro cloud research by addressing the challenge to keep data available in micro clouds. We propose a novel protocol, which encourages inter micro cloud communications. The core idea of the protocol is to allow cars in a different micro cloud to transfer the data of another micro cloud, which they brought with themselves while leaving, to those cars which will be later joining the same micro cloud. We also evaluate the protocol in many traffic densities to find the benefits of our protocol.

Our contributions can be summarized as:

- We introduce a novel inter micro cloud coordination protocol for stationary micro clouds, which relies upon

mobility of cars, i.e., cars forward the known data belonging to another micro cloud to cars approaching that micro cloud, so that the data lost from the micro cloud can be recovered (Section III).
- We enhance the protocol further by adaptively selecting the number of micro clouds in coordination with each other and also data transmission rates based upon current channel utilization (Section IV).
- In a detailed performance evaluation with varying car densities, we show the necessity and outline the benefits of inter micro cloud coordination compared to independent micro clouds with no coordination (Section V).

II. Related work

Eltoweissy et al. [4], Gerla [5], and Dressler et al. [6] introduced the concept of vehicular clouds with an aim to bring the cloud concept to vehicular networks. Lee et al. [11] discussed architecture and design principles of vehicular clouds from a systems, networking, and service perspective. The core idea is to organize vehicles to cooperatively use their storage, computing and networking resources for service provision [12].

Exploiting unused vehicular resources is also the foundation of Mobile Edge Computing, originating from *cloudlets* [13] and fog computing [14]. The key idea is to use resources at the edge of the network and therefore closer to the end-user. ETSI is currently in the process of standardizing an MEC architecture [3], [15] with potential applications like augmented reality and connected cars [15]. In a survey, Mach and Becvar [16] classified proposed MEC applications by outlining three service categories: consumer oriented services, operator and third-party services, and network performance and QoE improvement services.

MEC and vehicular clouds have been combined by Higuchi et al. [7]. By arranging *micro* and *macro* clouds, a large number of services can be offered. On one hand, micro clouds are usually small vehicular clouds [5], [6] offloading resources and providing locally relevant services. On the other hand, macro clouds are envisioned to span whole cities in order to provide services originating farther away or in a data center.

Recently, Hagenauer et al. [8] investigated a data collection application in vehicular micro clouds. They introduced algorithms to form micro clouds based upon geographic locations and a *dedicated* car chosen as a cluster head in the micro cloud is responsible for collecting data from other members and uploading it. We investigated the benefits of using multiple technologies to avoid overloading any specific channel for uploading data from micro clouds to data centers [17].

This paper focuses on vehicular micro clouds, which are usually formed using clustering concepts. Clustering groups cars based on similar parameters, e.g., position, or direction [18]. In the scope of this paper, we assume that micro clouds are placed at suitable locations and cars are aware of them [19].

Data in vehicular clouds is usually associated with specific geographic locations, thus, favoring ICN [9] and NDN [10] architectures. Quite some work has been done in this context for efficient data delivery and caching. In mobile ad-hoc networks, Bellavista et al. [20] proposed opportunistic resource replication middleware which is tolerant to node exit or failure events. For vehicular networks, Lee et al. [21] proposed exploiting vehicular mobility to diffuse sensed data. Amadeo et al. [22] designed a content-centric framework for vehicular networks on top of IEEE 802.11p protocol stack addressing reliable data delivery. Grassi et al. [23] presented an approach using road-topology to determine shortest paths to data residing locations for efficient fetching in vehicular NDN. Rao et al. [24] proposed prefetching and caching in NDN for the networks with user mobility when a handover is about to happen. Grewe et al. [25] defined popularity of data using to their request count and proposed prefetching of data according to their popularity.

Most recently, Grewe et al. [26] formulated cache capacities and probabilities to retrieve the data from the cache successfully. They also presented different caching strategies to improve cache utilization and increase the efficiency of data delivery in information-centric vehicular applications. To avoid excessive cache utilization, Hu et al. [27] also proposed controlled redundancy of data using erasure coding techniques. Higuchi et al. [28] proposed an algorithm that relies upon mobility of the vehicles to find an appropriate and small set of cars which should keep data copies so that the data can stay within its parent micro cloud for the maximum amount of time. Their idea is to select those set of vehicles whose mobility is not correlated, i.e., they are not moving in such a way that all of them leave the micro cloud in very short span of time.

While these works aim for efficient data diffusion and cache utilization when the data is present in the network, our designed protocol complements the existing works by proposing an adaptive protocol which aims to recover the data at certain geographic locations which was lost due to vehicular mobility.

III. Inter Micro Cloud Coordination

In this paper, our focus is on stationary vehicular micro clouds formed at an intersection. Each data content is associated with one vehicular micro cloud, which we call *parent micro cloud*. Our goal is to maximize the data availability within its parent micro cloud. The main challenge in this case is posed by mobility of cars. The cars join the micro cloud, gather some data in the micro cloud and leave. If the car leaving the micro cloud is the last one to have this data copy, then the data is lost from the parent micro cloud and it cannot be recovered.

Due to the inherent mobility in vehicular networks, data loss in micro clouds is inevitable. Our proposed protocol provides a solution to bring the lost data back to the parent micro cloud using neighboring micro clouds.

A. Prerequisites

For the protocol to work properly, the following assumptions are made.

- *Networking capabilities:* We assume that all cars are equipped with wireless networking radios. In the scope of this paper, we rely upon IEEE 802.11p for inter vehicle communications. However, this could also be replaced

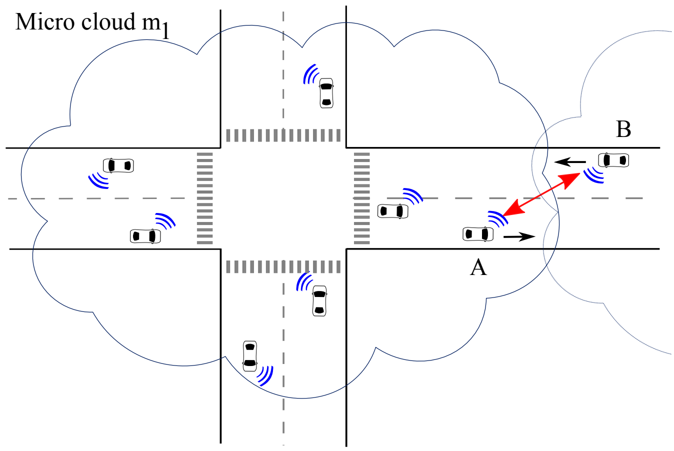

Figure 1. Typical micro cloud scenario: Car A leaves the micro cloud m_1 and car B is approaching m_1. A transfers the data contents of m_1 to B, thus, B brings back the data copies within the parent micro cloud m_1.

by other technologies as the protocol does not make any technology specific assumptions.

- *Geographic position and route:* We assume that all cars are equipped with a GPS device to know their current location and they also know the route which they are following.
- *Micro cloud knowledge:* All the cars need to be familiar with the location of micro clouds and their ids. This is an important information for a car as it needs to broadcast interest in data content of micro clouds. It is safe to assume that each car downloads information about all micro clouds (location, size, and id) along its followed route once before starting the trip.
- *Storage capabilities:* All cars are ready to offer their storage resources to micro cloud services.

B. Example Scenario

In Figure 1, we can see a typical micro cloud scenario. The micro cloud m_1 is established at an intersection. Car A, which is currently a member of m_1, is about the leave m_1. If A has a last copy of some data contents of m_1, then those data contents will get lost from m_1 as soon as A leaves. Car B, which is currently a member of another micro cloud, is going to join m_1 soon. It announces via control beacons that it is interested in data contents of m_1. When A receives a control beacon from B, it transfers the data contents of m_1 to B. Conversely, A announces its interest for data contents of the other micro cloud and receives the data from B, thus helping recover the lost data copies (if any) in both micro clouds.

C. Coordination Protocol

Our protocol is capable to function in a distributed fashion without any additional infrastructure support. We believe, deploying roadside units running the protocol can further improve the performance, but within the scope of this paper, we assume that there is no infrastructure available in the region. Figure 2 shows the proposed inter micro cloud coordination protocol in detail. There are two types of data exchange taking place in the protocol.

1) Control beacons: Each car periodically broadcasts control beacons. These beacons include (1) the sender car's id, (2) the current micro cloud id, (3) a set of data contents it has, and (4) a set of missing data content ids, which it desires to have access to. We refer to the set of missing data content ids as δ.

The control beacons are used by receiving cars to maintain a micro cloud metadata table. An example micro cloud metadata table for a car c_1 is shown in Table I. First row of the table is reserved for the car itself. We can see that c_1 is a member of micro cloud m_1 and it has two data items with ids d_1 and d_2, both of which are associated to micro cloud m_1. Car c_2 is also a member of m_1. It has only d_1 data content and is interested in d_2 and d_3. There is an interesting entry for car c_4. It is a member of micro cloud m_2 and has data content d_8 which is associated to micro cloud m_2. Note that it is interested in data contents d_1 and d_2 which belong to micro cloud m_1. One possible interpretation for this is that c_4 is soon going to leave m_2 and join m_1. It has data contents of m_2, but on receiving control beacons from c_1, it has also shown interest in the data from m_1 which is owned by c_1. So, d_1 and d_2 of m_1 are seen as missing data for c_4.

To generalize, let M be a set of next *n* micro clouds along the route of a car. It has micro cloud metadata table T with r rows. An i^{th} row $\forall i \in (0, r)$ in T is represented as $T[i]$. Let there be k_i data elements in the data set in $T[i]$ and j^{th} data element in $T[i]$ is given by $\{d_{T[i],j}, m_j\}$. The 0^{th} row in metadata table has the information about the car itself. So, the delta for the car is calculated as

$$\delta_{T[0]} = \{\bigcup_{i=1}^{r-1} \bigcup_{j=0}^{k_i} \{d_{T[i],j}, m_j\} \forall m_j \in M\} \setminus \bigcup_{j=0}^{k_0} \{d_{T[0],j}, m_j\}, \tag{1}$$

where $\bigcup_{j=0}^{k_0} \{d_{T[0],j}, m_j\}$ represents data contents owned by the car itself and $\bigcup_{j=0}^{k_i} \{d_{T[i],j}, m_j\} \forall m_j \in M$ represents the latest data contents associated to micro cloud $m_j \in M$ which are owned by another car whose entry is stored in i^{th} row in table T.

2) Micro cloud data: In the micro cloud metadata table, each car maintains the knowledge about data contents, which are accessible and missed by itself and other cars. In order to transfer the micro cloud data, the car calculates a set of data

Table I
AN EXAMPLE MICRO CLOUD METADATA TABLE FOR CAR c_1.

Car	M.Cloud	Data	Missing Data (δ)
c_1	m_1	$\{\{d_1, m_1\}, \{d_2, m_1\}\}$	$\{\{d_3, m_1\}\}$
c_2	m_1	$\{\{d_1, m_1\}\}$	$\{\{d_2, m_1\}, \{d_3, m_1\}\}$
c_3	m_1	$\{\{d_2, m_1\}, \{d_3, m_1\}$ $,\{d_1, m_1\}\}$	ϕ
c_4	m_2	$\{\{d_8, m_2\}\}$	$\{\{d_1, m_1\}, \{d_2, m_1\}\}$

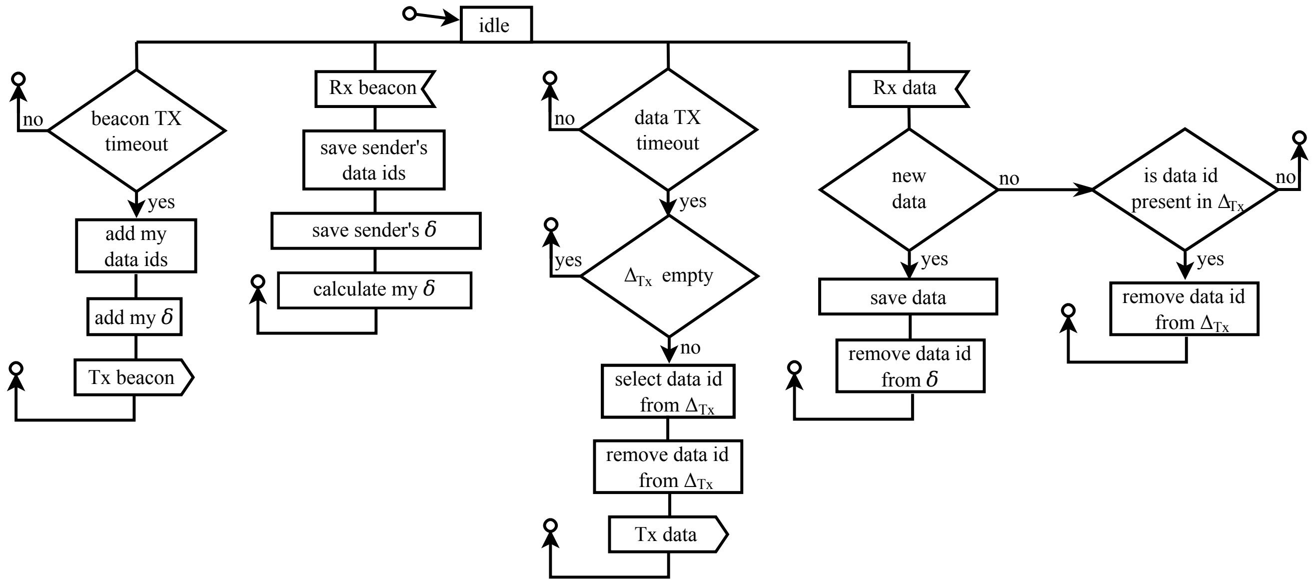

Figure 2. Details of the proposed inter micro cloud coordination protocol.

content ids which it has but others miss (referred to as Δ_{Tx}).

$$\Delta_{\text{Tx}} = \{\bigcup_{j=0}^{k_0}\{d_{T[0],j}, m_j\}\} \bigcap \{\bigcup_{i=1}^{r-1}\{\delta_{T[i]}\}\}, \tag{2}$$

where $\bigcup_{j=0}^{k_0}\{d_{T[0],j}, m_j\}$ represents the data contents accessible by the car, and $\bigcup_{i=i}^{r-1}\{\delta_{T[i]}\}$ represents all the δ missed by other cars in their control beacons. One data content is selected from the calculated Δ_{Tx} for transmission. Many factors can be taken into account for this selection, e.g., remaining lifetime of data before it gets outdated, timestamp of last transmission of the data id, approximate time for which some other car who also has the data copy is going to be present in the micro cloud. Considering these factors, some heuristic algorithm can be developed for advanced data selection.

In our protocol evaluations, we use the remaining lifetime of data as the deciding factor for the selection:

$$d_{\text{Tx}} = \operatorname*{argmax}_{d \in \Delta_{\text{Tx}}} \texttt{LIFETIME}(\texttt{d}), \tag{3}$$

where d_{Tx} is the data in Δ_{Tx} who has maximum remaining lifetime and $\texttt{LIFETIME}(d)$ gives the remaining lifetime of the data content d.

Taking the other factors into consideration for advanced selection is left for the future work. Whenever a car receives a data d_{Tx}, it saves the data, i.e. $\bigcup_{j=0}^{k_0}\{d_{\text{T}[0],j}, m_j\} \bigcup d_{\text{Tx}}$.

To avoid saturating the channel with the same data transfer by several cars, receiving d_{Tx} also triggers removal of same data from Δ_{Tx}, i.e.,

$$\Delta_{\text{Tx}} = \Delta_{\text{Tx}} \setminus d_{\text{Tx}}.$$

IV. Adaptive Inter Micro Cloud Coordination

Allowing cars to request data contents not just from their current micro cloud but also the next micro clouds along the route helps to recover the lost data back in micro clouds. However, there is a limit on the number of upcoming micro clouds for which it can effectively request data. This is because increasing data requests eventually leads to very frequent data exchange, which can quickly overload the wireless channel.

To overcome this problem, we designed an adaptive protocol, which takes current channel utilization into account. Each car periodically measures the channel utilization relying on the carrier sensing mechanism. The channel utilization is given by

$$\gamma = \frac{t_{\text{busy}}}{t_{\text{busy}} + t_{\text{idle}}} \ , \tag{4}$$

where t_{busy} is the time for which channel was sensed busy and t_{idle} is the time for which channel was sensed idle.

Let N be the maximum number of micro clouds (excluding the current micro cloud) for which a car is allowed to request data. Then, the number of micro clouds (excluding the current micro cloud) for which the car actually requests data based upon current channel utilization is given by

$$n = \begin{cases} N - \lfloor \gamma \times \omega \rfloor & \text{if } N > \lfloor \gamma \times \omega \rfloor \\ 1 & \text{if } N \leq \lfloor \gamma \times \omega \rfloor \end{cases} \ , \tag{5}$$

where ω is a scaling factor to optimize the n selection. In our evaluations, we assume ω to be 10 for the simplicity of converting $\lfloor \gamma \times \omega \rfloor$ to integers ranging in $[0, 10]$. However, further parameter studies can be conducted for finding the optimal values of ω, which is left for future work. It can be observed that as channel utilization γ increases, the car reduces the number of micro clouds for which it requests data. In case of very high channel utilization, where $N \leq \lfloor \gamma \times \omega \rfloor$, the car will request data for the current micro cloud and the next one.

Furthermore, the data transmission interval is also adaptable. We introduce a data transmission window which is given by

$$\text{Tx}_{\text{window}} = [0, \texttt{max}(\texttt{1}, \lfloor \gamma \times \omega \rfloor)] \ . \tag{6}$$

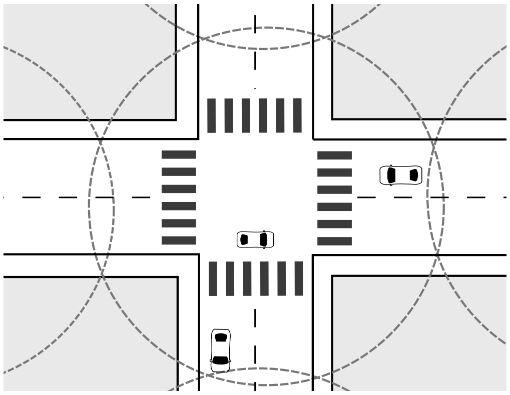

Figure 3. Micro clouds are established at each intersection of the Manhattan Grid. Dashed circles represent the micro cloud boundaries and gray squares represent buildings.

A random value within this Tx_{window} is selected as the next transmission interval. As the channel utilization increases, the window size increases as well, thus increasing the average data transmission interval which helps avoiding channel overloading.

V. Performance Evaluation

A. Simulation Setup

To evaluate the performance of our inter micro cloud coordination protocol, we used vehicular network simulation toolkit Veins [29], which couples road traffic simulator SUMO[1] with network simulator OMNeT++.[2]

We configured a Manhattan Grid scenario with different number of cars in simulation, i.e., 50, 75, 100 and 200. This allows us to evaluate the capabilities of our designed protocol in different vehicle densities. As shown in Figure 3, micro clouds are established at each intersection of the Manhattan Grid. In each micro cloud, a new data content is generated every 50 s. The life time of each data is 300 s. We also configure the cars to express their interest in only a certain percentage of the total data in micro cloud (25, 50, 75 and 100 %). Here, 25 % data interest means that at any point of time, a car is interested to request for only 25 % of the total micro cloud data. This parameter has been selected keeping in mind that some of the contents might not be relevant to a particular vehicle depending on a set of applications it runs on the on-board computer unit. From a user's perspective, the vehicle may not want to request data contents which are not relevant to it. This parameter comes in action with an additional effect on channel utilization due to increased number of data requests and higher data exchange transmissions with increasing data interests.

We compare several variants of our protocol:

- For $n = 0$, inter micro cloud coordination does not take place at all. This is used as a baseline. All the cars are interested in data belonging to the current micro cloud.
- For $n = 1$, the cars are interested in data belonging to current micro cloud and immediate next one. In other words, only the immediate neighboring micro clouds coordinate with each other.
- For $n = 2$, the cars are interested in data belonging to current and next two micro cloud along their route. In other words, three adjacent micro clouds coordinate with each other to maintain the data.
- For $n =$ dynamic, n is selected adaptively as explained in Section IV. In this case, we use $N = 6$, i.e., a car can request for data contents of current and at most next 6 micro clouds along its route. The exact value of n is selected based upon Equation (5).

[1]http://sumo.dlr.de

[2]http://www.omnetpp.org

Table II
Most relevant simulation parameters.

Parameter	Value
Channel	5.89 GHz
Transmission power	20 mW
Bandwidth	10 MHz
Data rate	6 Mbit/s
Building size	400 m × 400 m
Micro cloud radius	200 m
Manhattan Grid	3 × 3
New data generation interval	5 s
Data lifetime	300 s
Data size generated	1, 2, 4 and 10 kB
Control beacon interval	1 s
Cars	50, 75, 100 and 200
Data interest	25, 50, 75 and 100 %
ω in adaptive protocol	10
Repetitions per configuration	5
Simulation duration	1200 s

For all of the variants, control beacon interval is selected as 1 s. Micro cloud data transmission interval is also selected as 1 s, except for $n =$ dynamic, where the protocol adaptively selects the data transmission window (cf. Equation (6)). We rely upon IEEE 802.11p at a data rate of 6 Mbit/s in our simulations. However, the protocol is designed to operate irrespective of the communication technology being used.

To avoid any boundary effects in data collection, we recorded data in the central micro cloud which is surrounded by other micro clouds in all directions. Each simulation configuration was repeated 5 times with different seeds to improve statistical evidence. All relevant simulation parameters are summarized in Table II.

B. Lifetime of a Car in a Micro Cloud

First, we look into the time spent by cars in a micro cloud. This is an important metric because this gives us an immediate idea about how long a data can stay in the micro cloud if no data transmissions take place. Figure 4 shows the eCDF of time spent by cars in a micro cloud. On an average, a car is present in micro cloud for about 39 s. A very small percentage of cars which do not stop at the intersection cross the micro cloud in less than 10 s. Few cars stay in the micro cloud a little longer, but not more than 60 s. This gives us the clear picture of the challenges imposed by vehicular mobility to keep the data available in their parent micro cloud: All data needs to be replicated within this time frame.

C. Channel Utilization

Figure 5 shows the average channel utilization in all the configurations with 95% confidence intervals. Sometimes, the confidence intervals are too small, thus proving the statistical significance. In different configurations, average channel utilization ranges from nearly 5 % to as high as 70 %. This shows that we have studied the performance of our protocol in all scenarios ranging from negligible channel utilization to highly congested channel which is mainly because of the different data content sizes and different vehicular densities.

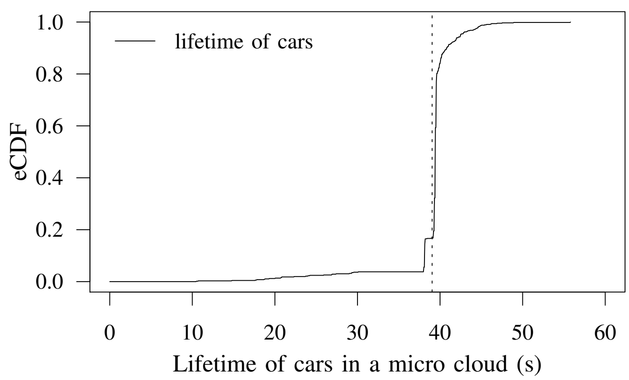

Figure 4. Lifetime of cars in a micro cloud (as observed in dense traffic scenario with 200 cars in simulation) plotted as an eCDF. The dotted vertical line at 39 s represents the average lifetime.

As n increases, cars start requesting for more data belonging to other micro clouds. This results in more data transfers and eventually, higher channel utilization. Interestingly, for n = dynamic, we can see that for high traffic density and 10 kB data size, channel utilization is much less than other configurations due to its adaptive design (cf. Section IV). Using IEEE 802.11p at 6 Mbit/s, the channel already gets overloaded with 10 kB data size in the 200 cars configuration. The protocol does not depend upon the technology in use and using other technologies with higher data rates, the protocol can even help maintaining data size greater than 10 kB.

D. Data Availability in Micro Cloud

Figure 6 shows the average time for which each data content is available in its parent micro cloud. The error bars represent 95% confidence intervals. Several interesting results can be observed in this plot. Firstly, for low and medium vehicle densities, i.e., 50 and 100 cars, respectively, the data content availability time in parent micro cloud increases as the value of n increases. Since, there is not enough channel load due to sparse vehicular density (cf. Figure 5), the dynamic n selection shows the best result as n becomes large (cf. Equation (5), with $N = 6$).

For dense vehicle density scenario, i.e., 200 cars, the average data contents availability time decreases for $n = 2$, and then increases again for n = dynamic. This shows that fixing

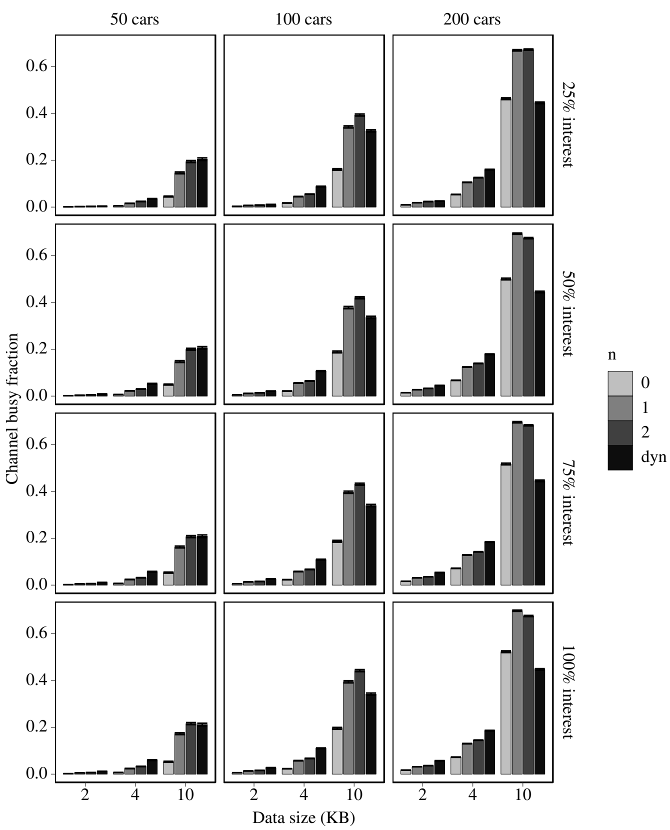

Figure 5. Average channel busy fraction observed in the micro cloud.

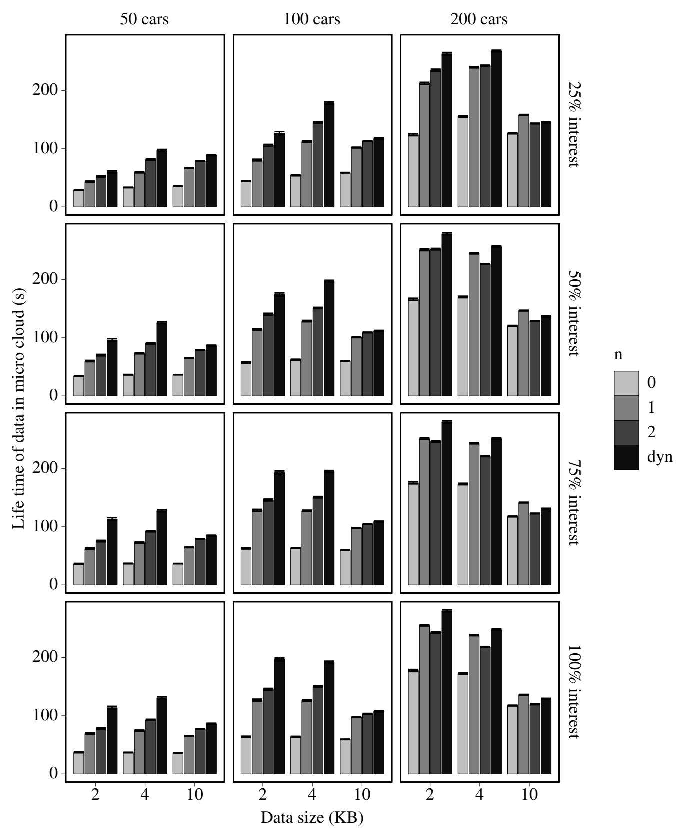

Figure 6. Average time for which each data is available in the micro cloud.

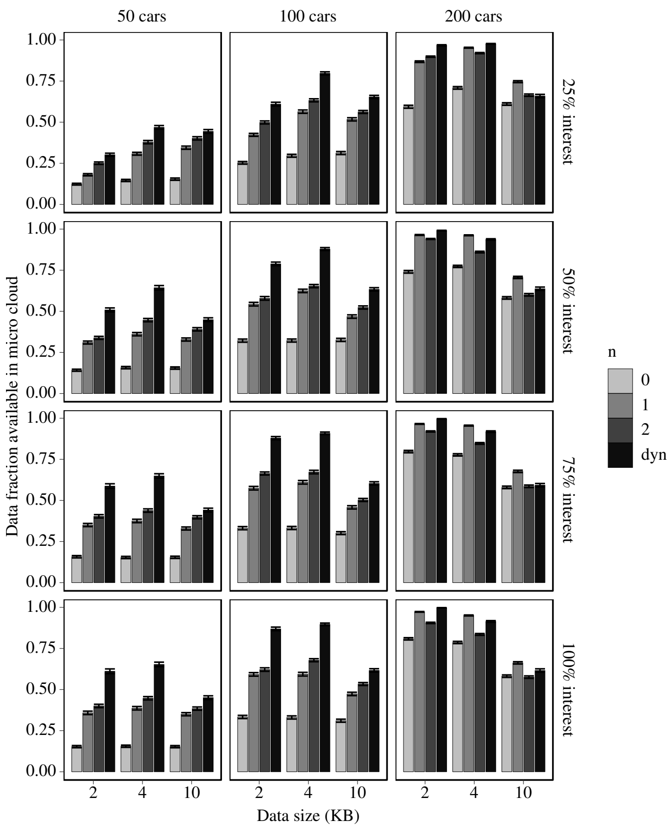

Figure 7. Average fraction of data available in the micro cloud.

n can have negative effects due to increased data exchange transmissions for more data contents and eventually, higher channel utilization.

Comparing the data size, we can see that as the size of data contents keeps on increasing, the average time of data availability is reduced. This is because the larger data contents are split into fragments and the data content is marked available in micro cloud only when all the fragments are present.

As the data interest increases from 25–100 %, the average time for data availability increases for low and medium traffic densities, but not for high traffic densities.

E. Fraction of Data Available in Micro Cloud

Another important metric that we study is fraction of data which is available it the micro cloud. It is calculated as:

$$\mathrm{f}_m = \frac{\text{Unique data present in micro cloud } m}{\text{Total unique data expected in micro cloud } m}. \quad (7)$$

To evaluate this metric, we sampled total unique data present in a micro cloud and total unique data expected to be present in the micro cloud periodically every second. Figure 7 shows the fraction of data available in different configurations with 95% confidence intervals. As the data interests increase from 25–100 %, for low traffic densities, more data can be maintained in parent micro clouds. We do not see any significant changes in the data fraction availability in micro cloud for medium and higher traffic densities.

For $n = 0$, cars are interested in the current micro cloud data only. As a result, for low and medium traffic densities where there are not enough cars in the micro cloud, the data loss is very frequent. By increasing n, we can recover the lost data and hence, more data can be maintained within parent micro cloud. This can be verified in the 100 cars, 100 % interest configuration. $n =$ dynamic helps in maintaining much higher data fraction when compared to $n = 0$.

For higher traffic densities, when there are plenty of cars in a micro cloud, $n = 0$ can also maintain around 70 % of the small sized data (2 and 4 kB). For $n = 1$, we are able to maintain more than 95 % of the data. The data fraction availability falls when n increases from 1 to 2. This is because increasing n also results in more data requests, which eventually leads to more transmissions and a higher channel utilization (cf. Figure 5). Under highly congested channel, less data is transmitted successfully due to increased collisions.

The adaptive protocol (cf. Section IV) tries to bring channel utilization under control by not just reducing the value of n but also by increasing the data transmission intervals. This helps in maintaining comparable data fractions for $n = 1$.

By maintaining larger amounts of data (cf. Figure 7) for significantly larger time (cf. Figure 6) compared to $n = 0$, vehicular micro clouds can be a good platform for applications like cooperative driving, intersection management, etc. even in low and medium traffic densities.

F. Redundancy of Data in Micro Cloud

Since our protocol works opportunistically without requiring any extra infrastructure, it is important to study the level of data redundancy in the micro cloud, i.e., on an average, how many data copies are present in a micro cloud. Figure 8 shows the average number of data copies in micro cloud with 95% confidence intervals. For low traffic density (50 cars), there are only 2 copies of data irrespective of data size or data interest percentage. This is traffic density dependent. As the traffic density increases, the number of data copies in micro cloud also increase.

An interesting trend is observed in 200 cars configuration. As the value of n increases, the average data copies in micro cloud decrease. This trend is observed because, when the neighboring micro clouds coordinate, there is a certain probability that the data exchange takes place outside the parent micro cloud. However, for $n =$ dynamic, the values are selected in real-time based upon current channel load and do not follow this trend.

Combining the observations from Figures 5 to 8, we can conclude that by enabling the coordination between neighboring micro clouds, the average lifetime of data in micro clouds can be improved significantly without significant increase in data redundancy and keeping the channel utilization under check.

G. Number of Transmissions per Data Content

Since cars are interested in the data contents of other micro clouds, we also measure the total transmissions for each data inside and outside the parent micro cloud. Figure 9 shows the average transmissions per data content with 95% confidence

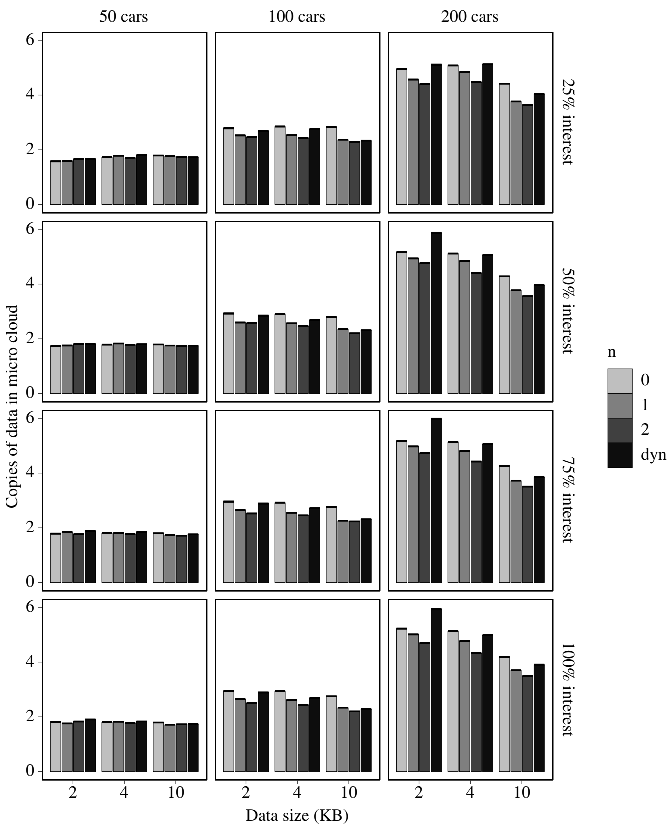

Figure 8. Average number of data copies in micro cloud.

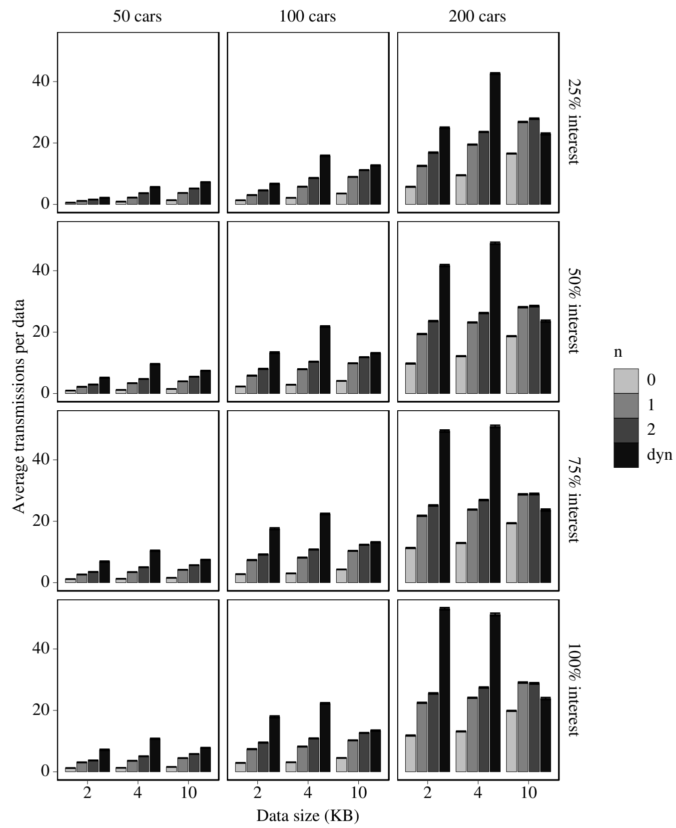

Figure 9. Average number of transmissions for each data in micro cloud.

intervals. As expected, $n = 0$ shows the minimum number of average transmissions per data content. As n increases, the transmissions increase as well.

For n = dynamic, the transmissions increased for data sizes of 2 and 4 kB. However, for 10 kB data size, when the channel gets saturated, the protocol adapts to lower the channel utilization by reducing the n value and also increasing the transmission interval window size.

VI. Conclusions

In this paper, we presented an inter micro cloud coordination protocol which aims at keeping data contents available in their parent micro cloud. A micro cloud is a small cluster of cars which acts as a virtual edge server offering caching and computational resources. In this context, it becomes important to keep the current data within its parent micro cloud, but it is often challenged by the mobility of cars. Cars join a micro cloud, collect some data, and contribute to the micro cloud services. As soon as they leave, the data may get lost.

Our protocol encourages cars to request data contents for not just the current micro cloud, but also for the micro clouds which are located along their followed route. This helps in recovering the lost data in micro clouds.

The performance of the protocol has been evaluated in a Manhattan grid scenario with various vehicle densities and also when cars request for only a certain percentage of the total data in micro cloud. From the obtained results, we can conclude that it is absolutely necessary to request for data contents of next micro clouds along the route. However, the number n of micro clouds for which the cars should request data is very critical. If n becomes too large, the channel gets congested. If it is very small, the benefits are not very significant. Our protocol takes into account the channel utilization and selects an appropriate value of n. As benefits, the protocol keeps channel utilization under control and also helps in keeping the data contents available within their parent micro cloud.

References

[1] O. Altintas, S. Banerjee, F. Dressler, and G. Heijenk, "Executive Summary - Inter-Vehicular Communication Towards Cooperative Driving," in *Dagstuhl Seminar 18202 - Inter-Vehicular Communication Towards Cooperative Driving*, Schloss Dagstuhl, Wadern, Germany: Schloss Dagstuhl, May 2018, pp. 31–59.

[2] S. Kim, B. Qin, Z. J. Chong, X. Shen, W. Liu, M. H. Ang, E. Frazzoli, and D. Rus, "Multivehicle Cooperative Driving Using Cooperative Perception: Design and Experimental Validation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 663–680, Apr. 2015.

[3] "Mobile Edge Computing (MEC), Framework and Reference Architecture," ETSI, GS MEC 003 V1.1.1, Mar. 2016.

[4] M. Eltoweissy, S. Olariu, and M. Younis, "Towards Autonomous Vehicular Clouds," in *2nd International ICST Conference on Ad Hoc Networks (ADHOCNETS 2010)*, J. Zheng, D. Simplot-Ryl, and V. C. M. Leung, Eds., Victoria, BC, Canada: Springer, Aug. 2010, 1–16.

[5] M. Gerla, "Vehicular Cloud Computing," in *11th IFIP/IEEE Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net 2012)*, Ayia Napa, Cyprus: IEEE, Jun. 2012, pp. 152–155.

[6] F. Dressler, P. Handle, and C. Sommer, "Towards a Vehicular Cloud - Using Parked Vehicles as a Temporary Network and Storage Infrastructure," in *15th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2014), ACM International Workshop on Wireless and Mobile Technologies for Smart Cities (WiMobCity 2014)*, Philadelphia, PA: ACM, Aug. 2014, pp. 11–18.

[7] T. Higuchi, J. Joy, F. Dressler, M. Gerla, and O. Altintas, "On the Feasibility of Vehicular Micro Clouds," in *9th IEEE Vehicular Networking Conference (VNC 2017)*, Torino, Italy: IEEE, Nov. 2017, pp. 179–182.

[8] F. Hagenauer, C. Sommer, T. Higuchi, O. Altintas, and F. Dressler, "Vehicular Micro Clouds as Virtual Edge Servers for Efficient Data Collection," in *23rd ACM International Conference on Mobile Computing and Networking (MobiCom 2017), 2nd ACM International Workshop on Smart, Autonomous, and Connected Vehicular Systems and Services (CarSys 2017)*, Snowbird, UT: ACM, Oct. 2017, pp. 31–35.

[9] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A Survey of Information-Centric Networking," *IEEE Communications Magazine*, vol. 50, no. 7, pp. 26–36, Jul. 2012.

[10] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, k. claffy, P. Crowley, C. Papadopoulos, L. Wang, and B. Zhang, "Named Data Networking," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 5, pp. 66–73, Jul. 2014.

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

[12] A. Boukerche and R. E. De Grande, "Vehicular Cloud Computing: Architectures, Applications, and Mobility," *Computer Networks*, vol. 135, pp. 171–189, Apr. 2018.

[13] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The Case for VM-Based Cloudlets in Mobile Computing," *IEEE Pervasive Computing*, vol. 8, no. 4, pp. 14–23, Oct. 2009.

[14] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog Computing and Its Role in the Internet of Things," in *1st Workshop on Mobile Cloud Computing (MCC 2012)*, Helsinki, Finland: ACM, Aug. 2012, pp. 13–16.

[15] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile Edge Computing - A key technology towards 5G," ETSI, White Paper No. 11, Sep. 2015. [Online]. Available: https://www.etsi.org/images/files/ETSIWhitePapers/etsi_wp11_mec_a_key_technology_towards_5g.pdf.

[16] P. Mach and Z. Becvar, "Mobile Edge Computing: A Survey on Architecture and Computation Offloading," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1628–1656, Mar. 2017.

[17] G. S. Pannu, T. Higuchi, O. Altintas, and F. Dressler, "Efficient Uplink from Vehicular Micro Cloud Solutions to Data Centers," in *19th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2018)*, Chania, Greece: IEEE, Jun. 2018.

[18] C. Cooper, D. Franklin, M. Ros, F. Safaei, and M. Abolhasan, "A Comparative Survey of VANET Clustering Techniques," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 1, pp. 657–681, Feb. 2017.

[19] T. Higuchi, F. Dressler, and O. Altintas, "How to Keep a Vehicular Micro Cloud Intact," in *87th IEEE Vehicular Technology Conference (VTC2018-Spring)*, Porto, Portugal: IEEE, Jun. 2018.

[20] P. Bellavista, A. Corradi, and E. Magistretti, "REDMAN: an Optimistic Replication Middleware for Read-only Resources in Dense MANETs," *Pervasive and Mobile Computing*, vol. 1, no. 3, pp. 279–310, Sep. 2005.

[21] U. Lee, B. Zhou, M. Gerla, E. Magistretti, P. Bellavista, and A. Corradi, "Mobeyes: Smart Mobs for Urban Monitoring with a Vehicular Sensor Network," *IEEE Wireless Communications*, vol. 13, no. 5, pp. 52 –57, Oct. 2006.

[22] M. Amadeo, C. Campolo, and A. Molinaro, "CRoWN: Content-Centric Networking in Vehicular Ad Hoc Networks," *IEEE Communications Letters*, vol. 16, no. 9, pp. 1380–1383, Sep. 2012.

[23] G. Grassi, D. Pesavento, G. Pau, L. Zhang, and S. Fdida, "Navigo: Interest Forwarding by Geolocations in Vehicular Named Data Networking," in *16th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2015)*, Boston, MA: IEEE, Jun. 2015.

[24] Y. Rao, H. Zhou, D. Gao, H. Luo, and Y. Liu, "Proactive Caching for Enhancing User-Side Mobility Support in Named Data Networking," in *7th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS 2013)*, Taichung, Taiwan: IEEE, Jul. 2013.

[25] D. Grewe, S. Schildt, M. Wagner, and H. Frey, "ADePt: Adaptive Distributed Content Prefetching for Information-Centric Connected Vehicles," in *87th IEEE Vehicular Technology Conference (VTC2018-Spring)*, Porto, Portugal: IEEE, Jun. 2018.

[26] D. Grewe, M. Wagner, S. Schildt, M. Arumaithurai, and H. Frey, "Caching-as-a-Service in Virtualized Caches for Information-Centric Connected Vehicle Environments," in *10th IEEE Vehicular Networking Conference (VNC 2018)*, Taipei, Taiwan: IEEE, Dec. 2018.

[27] B. Hu, L. Fang, X. Cheng, and L. Yang, "Vehicle-to-Vehicle Distributed Storage in Vehicular Networks," in *IEEE International Conference on Communications (ICC 2018)*, Kansas City, MO, USA: IEEE, May 2018.

[28] T. Higuchi, G. S. Pannu, F. Dressler, and O. Altintas, "Content Replication in Vehicular Micro Cloud-based Data Storage: A Mobility-Aware Approach," in *10th IEEE Vehicular Networking Conference (VNC 2018)*, Taipei, Taiwan: IEEE, Dec. 2018.

[29] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," *IEEE Transactions on Mobile Computing*, vol. 10, no. 1, pp. 3–15, Jan. 2011.