How to Keep a Vehicular Micro Cloud Intact

Takamasa Higuchi*, Falko Dressler† and Onur Altintas‡

*Toyota InfoTechnology Center, Tokyo, Japan

†Paderborn University, Paderborn, Germany

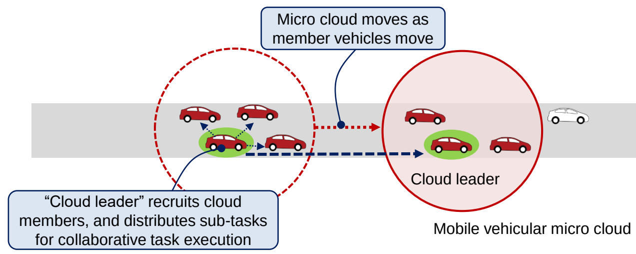
‡Toyota InfoTechnology Center, U.S.A., Mountain View, CA

***Abstract*—The emerging concept of vehicle cloudification is a promising solution to deal with ever-growing computational and communication demands of connected vehicles. A key idea is to have connected vehicles in the vicinity form a cluster, called *vehicular micro cloud*, and collaborate with other cluster members over vehicle-to-vehicle (V2V) networks to offer data processing, data storage, sensing and communication services. It allows us to use vehicles as virtual edge servers that complement traditional cloud and physical edge servers in the backbone network. In this paper, we design a mechanism to intelligently schedule where and when to form such vehicular micro clouds. A remote server maintains statistics of the amount of available on-board computational resources spatio-temporally, and analyzes these statistics to identify the locations where vehicles can consistently offer a sufficient amount of resources for service provisioning. The results from our proof-of-concept simulations show that our system can significantly reduce the risk of resource scarcity in vehicular micro clouds.**

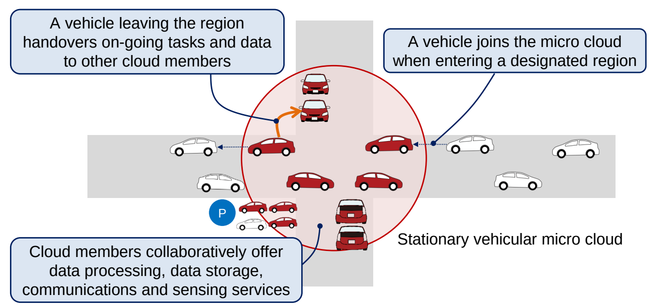
I. Introduction

Connected vehicles deal with an increasing amount of data to improve safety, efficiency and comfort of mobility. To date, vehicles typically rely on remote cloud servers to store, process and distribute data contents, ranging from digital maps to infotainment contents. In the long run, however, such a cloud-heavy architecture would be unlikely to scale well with the ever-growing service demands from connected vehicles, mainly because of the huge communication overhead to/from the remote servers. Although edge computing [1] would partly resolve the issue with the help of small-scale computational facilities (i.e., edge servers) in the backbone networks, vehicles still have to access the edge servers either by way of roadside units or via cellular radio access at the cost of non-negligible load on the radio access networks.

The emerging concept of *vehicle cloudification* [2] has the potential to address this issue. The idea is to have connected vehicles in the vicinity form a cluster, called a vehicular micro cloud, and collaborate with other cluster members over V2V networks to offer data processing, data storage, sensing and communication services. Depending on use cases, vehicular micro clouds can be formed in two different ways. A *mobile vehicular micro cloud* is formed by a group of close-by vehicles moving in the same direction, and moves following the member vehicles as shown in Figure 1a. In contrast, a *stationary vehicular micro cloud* is tied to a fixed geographical region (e.g., an intersection). As illustrated in Figure 1b, a vehicle entering a designated region joins a stationary micro cloud, and contributes part of their computational resources for collaborative task execution. When exiting the region, the vehicle handovers its on-going tasks and relevant data to any other cloud member(s), so that the micro cloud can continue to provide services.



(a) Mobile cloud in a freeway



(b) Stationary cloud around an intersection

Figure 1. Vehicular micro cloud concept

Processing and caching data at the vehicular micro clouds would mitigate the need for vehicles to access remote cloud/edge servers, reducing the load on radio access networks. For example, a stationary vehicular micro cloud at a road intersection may maintain a 3-dimensional map of the local geographical region, containing information of dynamic objects like vehicles and pedestrians. The map can be updated on the spot by collecting sensor measurements from neighboring vehicles over V2V networks, followed by analyzing them in a distributed manner using on-board computers of micro cloud members. Automated driving vehicles in the region may download the up-to-date map information without accessing remote cloud or physical edge servers over cellular networks. Thus vehicular micro clouds can be used as virtual edge servers that assist traditional cloud and edge servers.

In this paper, we design and evaluate the *vehicular micro cloud planner*, a mechanism to intelligently schedule where and when to form the stationary type of vehicular micro clouds. A key challenge of maintaining stationary micro clouds

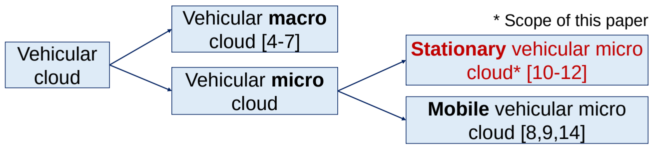


Figure 2. Classification of vehicle cloudification approaches

is that vehicles frequently join and leave the cluster over short periods of time. Consequently, the amount of computational resources that are available in a vehicular micro cloud can significantly vary at each moment. If the micro cloud fails to keep offering a sufficient amount of resources to host on-going computational services, the execution state of programs and relevant data might be lost, or have to be handed over to an alternative computational entity (e.g., remote servers or another vehicular micro cloud). In order to avoid such potential performance degradation and communication overhead due to task handovers, micro clouds can be formed at the geographical regions where connected vehicles are consistently available. To this end, our vehicular micro cloud planner maintains statistics of the amount of available on-board resources spatio-temporally, and analyzes these statistics to identify the most suitable position to set up a stationary micro cloud. The results from our proof-of-concept simulations show that the micro cloud planner can significantly reduce the risk of resource scarcity in vehicular micro clouds.

II. Related Work

The basic concept of vehicle cloudification has been established already in the literature. A common idea is to inter-connect on-board computers of connected vehicles over vehicular networks, so that they can collaboratively offer computational and data communication services to other vehicles and/or third party users (e.g., smartphone users, public authority, etc.). The existing solutions for vehicle cloudification can be classified as in Figure 2.

A *vehicular macro cloud* puts all the vehicles that are reachable over V2V networks into a single, city-scale vehicular cloud [3]–[5]. Vehicles offer their resources in the form of *services*, while users find appropriate service providers in the vehicular network using a service discovery platform. STAR [6] utilizes V2I communications to assist service discovery. Vehicles register available services to close-by roadside units, which maintain a list of services and potential service providers (i.e., service directory). When a service is requested by a user, roadside units look up their service directory to find a match. Car4ICT [3] employs fully distributed service discovery where vehicles collaboratively maintain the service directory over V2V networks.

In contrast, a vehicular micro cloud is a small-scale cluster, consisting of vehicles within a limited geographical region. Vehicles in a micro cloud can reach other cloud members within a small number of hops of V2V communications, which allows much tighter collaboration among member vehicles. As discussed in Section I, vehicular micro clouds can be further categorized according to their mobility. Gerla [7], Dressler [8] and Lee et al. [9] established a fundamental procedure to form mobile vehicular micro clouds. A vehicle that needs extra computational resources becomes a so-called *cloud leader*, and initiates a new micro cloud by inviting neighboring vehicles as cloud members. The leader then splits its application task into smaller sub-tasks and distributes them to the cloud members. Once the task execution is completed, each cloud member replies with the results, so that the leader can assemble them into the final output. Hagenauer et al. [10] proposed the idea of using stationary vehicular micro clouds as virtual edge servers that facilitate efficient collection of vehicle-generated data (e.g., measurements from on-board sensors) to remote cloud servers. Vehicles entering a pre-assigned geographical region join the corresponding vehicular micro cloud, and one of the cloud members serves as a cloud leader. The cloud members send their own data to the cloud leader over V2V networks, instead of individually uploading them to the remote servers via cellular radio access. The cloud leader then aggregates data from multiple cloud members before uploading them to the server to mitigate the load on cellular networks. Some other works have examined the possibility of forming stationary vehicular micro clouds by parked cars [11], [12]. On-board computers are mostly idle while being parked, giving them chance to offer rich amount of computational resources to vehicular micro clouds. In addition, the stationary nature of parked cars makes them fit well in keeping data contents in a certain geographical region, or supporting data forwarding in V2V networks as virtual roadside units. The authors recently investigated the feasibility of both stationary and mobile vehicular micro clouds under practical road traffic conditions [2]. Analyzing realistic vehicle probe datasets from the city of Luxembourg [13] and a major inter-city highway in Japan [5], they show that stable vehicular micro clouds that consistently contain a sufficient amount of vehicles can be formed at many places throughout the road network.

In this paper, we focus on stationary vehicular micro clouds and investigate a mechanism to form stable micro clouds that can consistently offer sufficient amount of computational resources for a long period of time. Although most of the prior works on stationary micro clouds assume a control entity (e.g., cloud server) that pre-assigns geographical regions to form micro clouds, few of them have tackled the question of how to identify the most suitable regions to set up vehicular micro clouds. One of the most relevant work in the literature is the concept of software-defined vehicular clouds [14], where a backend server (i.e., VC controller) collects information of individual vehicles to maintain a comprehensive view of their mobility and resource availability. When the VC controller is requested to execute a set of computational tasks, it selects appropriate vehicles to assign each task, taking their predicted paths and available resources into account. However, the paper does not specify how to determine optimal set of vehicles to delegate computational tasks. To the best of our knowledge, we are the first to propose specific algorithms to intelligently schedule formation of vehicular micro clouds.

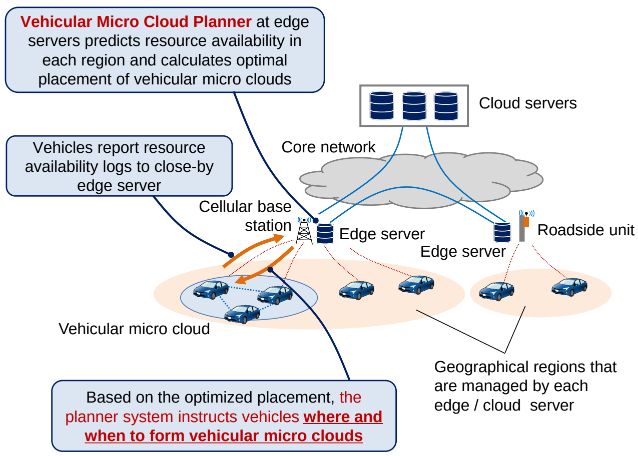


Figure 3. Architecture of vehicular micro cloud planner

III. Architecture

Figure 3 illustrates general architecture of the vehicular micro cloud planner. Vehicles report availability of computational resources in their on-board units (e.g., CPU cycles, data storage, memory space, etc.) to a close-by edge server, which hosts the vehicular micro cloud planner. The resource availability logs are geo-tagged and timestamped, so that the planner system can maintain history of resource availability at each location across a certain geographical region.

We assume that a system operator (e.g., traffic authority) specifies a set of candidate geographical regions to form vehicular micro clouds. The system operator may also specify multiple sets of candidate regions to set up multiple micro clouds. At each time slot, the planner system selects the most suitable candidate region based on the current and/or statistical resource availability information, and notifies vehicles of its decision. The vehicles entering the designated region initiate a procedure to join the vehicular micro cloud and contribute their on-board resources to offer computational services. The maintenance of vehicular micro cloud is beyond the scope of this paper, and we leave it for our future work.

IV. Lifetime-based Micro Cloud Planning

A key component of the vehicular micro cloud planner is an algorithm to select a region to set up a micro cloud from the given set(s) of candidate regions $\mathcal{C}$. While there can be a variety of possible criteria for the selection, in this work we focus on reducing the frequency of *failure events*, where the amount of available computational resources in an active vehicular micro cloud drops below a pre-defined threshold. As discussed in Section I, such failures often cause the risk of losing data contents and/or communication overhead for task handovers. For simplicity of discussion, we assume that the amount of available resources in a vehicular micro cloud is proportional to the number of vehicles in the designated geographical region, and a failure event occurs when the

Algorithm 1 Lifetime-based micro cloud planning

for each time slot **do**
  if the number of member vehicles in an existing micro cloud drops below N **then**
    release the micro cloud
  end if
  if no micro cloud is formed at any regions in $\mathcal{C}$ **then**
    $\mathcal{C}' \leftarrow$ candidate regions with more than N vehicles
    if $\mathcal{C}' \neq \phi$ **then**
      for each candidate region $c \in \mathcal{C}'$ **do**
        predict lifetime of potential micro cloud at c
      end for
      $\tilde{c} \leftarrow$ the region w/ the longest expected lifetime
      if (expected lifetime at $\tilde{c}$) $\geq l_{min}$ **then**
        set up a new micro cloud at region $\tilde{c}$
      end if
    end if
  end if
end for

number of vehicles in a micro cloud drops below N. Unless otherwise noted, we set N to 10 in this paper.

Algorithm 1 outlines the procedure of our planner system. The planning process is initiated when the existing micro cloud can no longer offer sufficient amount of computational resources (*i.e.*, failure event). At the beginning of the process, the vehicular micro cloud planner predicts the period of time for which each candidate region in $\mathcal{C}$ consistently contains no less than N vehicles. This time period can be interpreted as the expected lifetime in the case that a micro cloud is set up in the candidate region. The planner then identifies the candidate region $\tilde{c} \in \mathcal{C}$ that has the longest expected lifetime. To suppress frequent failures, the planner sets up a new micro cloud at $\tilde{c}$ only if its expected lifetime is no less than a pre-defined threshold l_{min}. We set l_{min} to 15 seconds in this paper.

V. Evaluation

We have conducted a proof-of-concept simulation using a road traffic simulator SUMO [15] to evaluate basic feasibility of the vehicular micro cloud planner.

A. Scenario and Assumptions

We employ Luxembourg SUMO traffic scenario [13] to simulate realistic road traffic for 24 hours in the city of Luxembourg. The road traffic is generated based on the statistical traffic demands derived from censuses, while the SUMO simulates micro behavior of individual vehicles. The number of vehicles in the simulation amounts for more than 5,000 during peak hours. We recorded locations of all these vehicles every second to obtain a synthetic probe dataset.

We assume that a system operator provides the planner system with 7 sets of candidate regions $\mathcal{C}_1, \ldots, \mathcal{C}_7$ to set up 7 vehicular micro clouds. A candidate region in $\mathcal{C}_i$ is defined as a circular region with a radius of 300 meters. Each candidate

Algorithm 2 Density-based micro cloud planning (baseline)

```
for each time slot do
    if the number of member vehicles in an existing micro cloud drops below N then
        release the micro cloud
    end if
    if no micro cloud is formed at any regions in 𝒞 then
        𝒞′ ← candidate regions with more than N vehicles
        if 𝒞′ ≠ φ then
            c̃ ← the region w/ the max. number of vehicles
        end if
    end if
end for
```

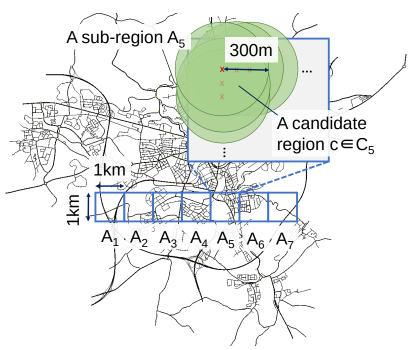


Figure 4. Simulation field

set $\mathcal{C}_i$ contains 10,000 candidate regions, which are uniformly distributed over a $1\,\mathrm{km} \times 1\,\mathrm{km}$ sub-region A_i (cf. Figure 4).

For simplicity of discussion, we assume that the vehicular micro cloud planner can predict the lifetime of micro clouds with no error based on the history of resource availability information. While it is an optimistic assumption, the simulation results would give us an insight into the potential of the lifetime-based micro cloud planning.

As a baseline for performance evaluation, we also evaluate density-based micro cloud planning, which is outlined in Algorithm 2. It is similar to the lifetime-based scheme except that it selects the candidate region with the largest instantaneous number of vehicles regardless of the expected lifetime.

B. Simulation Results

Figure 5 shows the average number of failure events in an hour for the sub-regions $A_1, \ldots, A_7$. The density-based cloud planner (referred to *Density-based*) suffers from frequent failure events, implying that a region containing a large number of vehicles may not necessarily be a suitable location to maintain a stable vehicular micro cloud. In contrast, our *Lifetime-based* solution can reduce the failures of micro clouds by 53-67% by taking the expected lifetime of potential micro clouds into account when selecting a region to set up a new micro cloud.

Meanwhile, the lifetime-based cloud planner could degrade the availability of micro clouds, as it filters out short-lived micro clouds to avoid frequent failures. To identify the negative impact of filtering, we have also examined the ratio

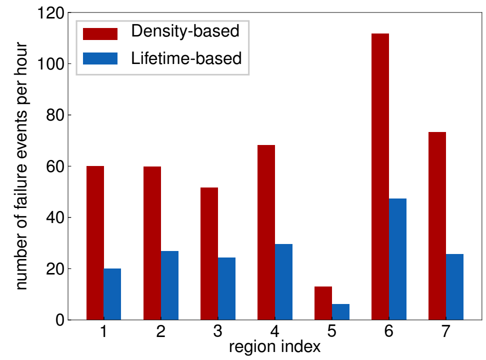


Figure 5. Number of failure events

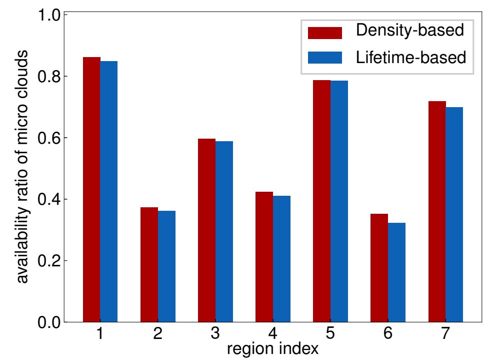


Figure 6. Availability ratio

of time when a micro cloud is available in each sub-region (i.e., availability ratio). The results in Figure 6 show that degradation of the availability ratio is up to a few percent. It indicates that the significant reduction of failure events can be achieved with little impact on micro cloud availability.

VI. Discussion

As final remarks, we discuss the open challenges that should be addressed in the next step.

A. Communication Overhead for Statistics Collection

We have assumed that the vehicular micro cloud planner collects resource availability logs from vehicles at the cost of a certain amount of communication overhead. Thus comparing the trade-off between the overhead of transmitting resource availability logs to the server with the overhead of handing over data and tasks from a dissolving vehicular micro cloud to another computational entity would be an important step to validate the soundness of this approach. In addition, contents and transmission frequency of resource availability logs would be also key factors that may affect performance and efficiency of vehicular micro clouds.

B. Combination of Multiple Criteria

In this paper, we have designed an example micro cloud planner that selects a candidate region with the longest expected lifetime, aiming to minimize failures of micro clouds. Of course, there can be a variety of other possible policies for micro cloud planning. Examples of such alternatives include combination of both the lifetime-based and density-based planning mechanisms. The micro cloud planner may first filter out the candidate regions whose expected lifetime is no more than a designated threshold (e.g., 1 hour), and then pick up a candidate region whose expected vehicle density during its lifetime is the largest among the remaining candidates. It would achieve a good trade-off between stability and computational power of vehicular micro clouds. The system operator should carefully select the most suitable policy depending on application requirements. Vehicle speeds and network conditions around candidate regions would also be possible factors to consider, as they may affect stability of micro clouds.

C. Predicting Resource Availability

A key technological component of the lifetime-based micro cloud planner is a mechanism to predict resource availability at each candidate region. The system should take both vehicle mobility and available resources in individual vehicles into account to make reasonable predictions. Traffic flow prediction in road networks has been one of the major research challenges in traffic engineering, and thus extensively investigated in the literature. We envision that traditional traffic prediction frameworks like ARIMA (Autoregressive, Integrated and Moving Average) [16] and Seasonal ARIMA [17] can be partly used to predict resource availability in potential vehicular micro clouds.

D. Predictive Handover

The algorithm in Section IV assumes that the micro cloud planner attempts to set up a new vehicular micro cloud when the available computational resources in an existing micro cloud drop below a designated lower bound. In order to minimize the risk of losing data contents and execution state of programs, however, the micro cloud has to complete handing over data and tasks to another computational entity before its own computational resources become scarce. Thus the micro cloud planner needs to set up an alternative micro cloud sufficiently prior to failure of the existing micro cloud and initiate a handover process. When to trigger a handover is another challenging research question, as too aggressive handovers would naturally increase the communication overhead. Accurate resource availability prediction also plays a key role for such predictive handover decisions.

VII. Conclusion

In this paper, we have designed the vehicular micro cloud planner, a mechanism to intelligently schedule formation of stationary micro clouds based on statistical knowledge about road traffic conditions and resource availability of on-board computers. The simulation results show that the risk of failures can be significantly reduced by scheduling micro clouds based on their expected lifetime. Along with the series of open issues discussed in Section VI, efficient management of data contents in vehicular micro clouds also constitutes a key technological challenge. The notion of information centric networking may hold promise as the basis of possible solutions [9].

References

[1] 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.

[2] 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.

[3] O. Altintas, F. Dressler, F. Hagenauer, M. Matsumoto, M. Sepulcre, and C. Sommer, "Making Cars a Main ICT Resource in Smart Cities," in *34th IEEE Conference on Computer Communications (INFOCOM 2015), International Workshop on Smart Cities and Urban Informatics (SmartCity 2015)*. Hong Kong, China: IEEE, Apr. 2015, pp. 654–659.

[4] X. Hou, Y. Li, M. Chen, D. Wu, D. Jin, and S. Chen, "Vehicular Fog Computing: A Viewpoint of Vehicles as the Infrastructures," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 6, pp. 3860–3873, Jun. 2016.

[5] F. Hagenauer, C. Sommer, R. Onishi, M. Wilhelm, F. Dressler, and O. Altintas, "Interconnecting Smart Cities by Vehicles: How feasible is it?" in *35th IEEE Conference on Computer Communications (INFOCOM 2016), International Workshop on Smart Cities and Urban Computing (SmartCity 2016)*. San Francisco, CA: IEEE, Apr. 2016, pp. 788–793.

[6] K. Mershad and H. Artail, "Finding a STAR in a Vehicular Cloud," *IEEE Intelligent Transportation Systems Magazine*, vol. 5, no. 2, pp. 55–68, Apr. 2013.

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] ——, "Parked Cars as Virtual Network Infrastructure: Enabling Stable V2I Access for Long-Lasting Data Flows," 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. 57–64.

[12] S. Arif, S. Olariu, J. Wang, G. Yan, W. Yang, and I. Khalil, "Datacenter at the Airport: Reasoning about Time-Dependent Parking Lot Occupancy," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 11, pp. 2067–2080, Nov. 2012.

[13] L. Codeca, R. Frank, and T. Engel, "Luxembourg SUMO Traffic (LuST) Scenario: 24 Hours of Mobility for Vehicular Networking Research," in *7th IEEE Vehicular Networking Conference (VNC 2015)*. Kyoto, Japan: IEEE, Dec. 2015.

[14] I. Jang, S. Choo, M. Kim, S. Pack, and G. Dan, "The Software-Defined Vehicular Cloud: A New Level of Sharing the Road," *IEEE Vehicular Technology Magazine*, vol. 12, no. 2, pp. 78–88, Jun. 2017.

[15] D. Krajzewicz, G. Hertkorn, C. Rössel, and P. Wagner, "SUMO (Simulation of Urban MObility); An Open-source Traffic Simulation," in *4th Middle East Symposium on Simulation and Modelling (MESM 2002)*, Sharjah, United Arab Emirates, Sep. 2002, pp. 183–187.

[16] S. Lee and D. Fambro, "Application of Subset Autoregressive Integrated Moving Average Model for Short-Term Freeway Traffic Volume Forecasting," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1678, pp. 179–188, 1999.

[17] B. M. Williams and L. A. Hoel, "Modeling and Forecasting Vehicular Traffic Flow as a Seasonal ARIMA Process: Theoretical Basis and Empirical Results," *Journal of transportation engineering*, vol. 129, no. 6, pp. 664–672, Nov. 2003.