Dynamic Resource Management of Real-Time Edge Services for Intelligent Vehicular Networks: A Case Study

Katja Gilly[1], Sonja Filiposka[2], Salvador Alcaraz Carrasco[1], Anastas Mishev[2]
[1]*Department of Computer Engineering, Miguel Hernandez University,*
Avenida de la Universidad de Elche, s/n, 03202 Elche, Alicante, Spain
[2]*Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University*
Rugjer Boshkovikj 16, 1000 Skopje, North Macedonia
katya@umh.es

***Abstract*–The Internet of Vehicles requires high bandwidth and low latency services to unleash the potential of fully connected vehicles. Thus, the offloading proposals that successfully manage massive real-time service requests from vehicle nodes are needed. In this paper, we analyse the dynamic resource management for intelligent vehicular networks based on Multi-access Edge Computing architecture services. Using a combination of CloudSim and SUMO simulators, we present a case study of infotainment services in the city centre of Alicante, in Spain, that shows a high degree of optimality both in service allocation and migration when considering dense urban environments.**

***Index Terms*–Edge computing; Resource management; Intelligent vehicles; 5G mobile communication.**

I. INTRODUCTION

The advances in intelligent vehicular networks [1] in combination with the new capabilities of 5G mobile radio systems [2] offering high bandwidth with very low latency have created unique opportunities for proliferation of new services accessible via the vehicle to infrastructure network ecosystem. Taking advantage of the computing power available at the edge of the network as described with the Multi-Access Edge Computing (MEC) architecture [3], a new class of 5G applications has emerged supporting the specific requirements of vehicular networks. These services include fully or highly autonomous driving with high-definition real-time maps and traffic monitoring, and richer passenger experience using infotainment systems [4]. Fully connected vehicles have very challenging requirements that can be fulfilled only by integrating them into the 5G networking landscape, where edge computing plays the key role as a supporting technology that enables hosting of real-time services for the connected vehicles. The service environment is running in the edge cloud-computing capabilities within the access network infrastructure in close proximity to vehicles.

All of these services require substantial computing and storage capabilities in combination with real-time responses. Since the scarce resources available in the vehicle are not enough, the services need to be hosted in the MEC environment using the resources available in the edge servers co-located with the 5G base stations. The requirements for real-time responses demand that the services must be hosted in the closest to the vehicle edge servers, i.e., located at the base station currently in use by the vehicle. The movements of the vehicles equipped with vehicle to everything (V2X) systems render the location of the initially used edge host non-optimal in the long run, even though the underlying network maintains the service continuity between the endpoints. For the MEC system to maintain the service requirements in a mobile environment, service mobility is required.

Therefore, it is of great importance that an efficient resource management framework is deployed in the MEC ecosystem, such that will enable constant maintenance of high quality of service for each hosted service.

In this paper, we describe a proposal for a dynamic resource management solution that provides the required efficiency when hosting services necessary for the implementation of intelligent vehicular networks applications such as autonomous driving. The proposed solution provides highly efficient placement and migration decisions that maintain high quality of service in terms of guaranteed computing and storage resources coupled with low latency. We present the efficiency of the proposed solution using a case study of offering real-time touristic infotainment services to vehicles around city attractions.

The rest of the paper is organised as follows. In Section II, the MEC architecture and the proposed dynamic resource management solution are described. In Section III, the case study of vehicular networks is presented. Section IV discusses the performance results from the case study and finally, section V concludes the paper.

II. DYNAMIC RESOURCE MANAGEMENT FOR MEC SERVICES

The traditional computing techniques that rely on the cloud for additional storage and computing resources are not suitable for time sensitive services that do not tolerate the high and sometimes unpredictable latency that accompanies

Manuscript received 19 October, 2018; accepted 3 May, 2019.
This research was partially funded by the NOA project from the Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University.

cloud hosted services. On the other hand, many newly developed Internet of Things services, and especially vehicle related services, require not only computing capabilities, but also very low latency so that (near) real-time responses can be produced. This has led to the development of the edge computing paradigm that brings the cloud computing capabilities to the edge of the provider network [5].

Within the edge computing architecture, the edge network elements, such as base stations for the mobile network providers, are equipped with a small number of edge nodes (servers) that can host the services requested by the mobile user equipment (Fig. 1). All resources available at the edge servers are virtualised, creating a pool of virtual resources available for allocation to multiple tenants. The cloud is used for two purposes:

– as a last resort, in the case when there are not enough resources in the edge servers to accommodate all service requests;

– for batch computing, such as machine learning, algorithms training or business analytics.

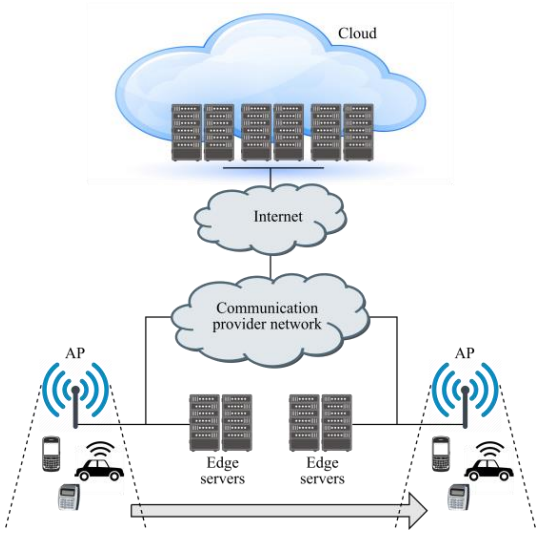

Fig. 1. Edge computing architecture.

However, to realise edge computing, the challenge of managing edge nodes needs to be addressed [6]. This includes both the problem of optimal placement of newly requested services and the continuous migration of the services based on the mobility pattern of the users. For these purposes, all edge servers in the providers' network are part of a separate 5G network slice, leveraging the Software Defined Network paradigm [7] to define a high-bandwidth virtual network that can be used for services migration. Using network slicing and service specific dynamic assignment, the available computing and storage virtualised resources can be offered to multiple simultaneous service requests in the vehicular network.

Each vehicle can request one or more services to be hosted at the edge, providing information about the computational requirements of the service, such as CPU, RAM, bandwidth, storage, etc., based on the required resources, the location of the vehicle, and the status of the currently allocated resources.

In order to provide the most optimal allocation of resources, such that the chosen server to host the vehicular service is as close to the vehicle as possible, we propose that the pool of virtual resources is described using a hierarchical tree. Within the hierarchical tree, the edge server virtual resources are represented as leaves, and they are grouped together on higher levels of the tree based on the network links that interconnect them. To obtain the hierarchical tree, a community detection algorithm [8] can be used with the 5G slice describing the virtual network that interconnects all edge servers in the provider network. In this way, each higher level of the tree represents a grouping of the resources available in the lower level.

In addition to the tree representation, the dynamic resource management system is comprised of several modules that keep track of the information needed to make optimal decisions: mobility tracker, quality of service (QoS) analyser, and VM manager. The mobility tracker is used to track the location of the vehicles and trigger the QoS analyser every time a handover occurs while the vehicle leaves the service area of the current base station. On every trigger, the QoS analyser checks the service parameters and starts the process of service migration if conditions are suboptimal. The VM manager module implements the placement and migration policies and manages the locations of the VMs on the MEC servers.

The actions taken in all modules work in concert to implement the dynamic resource management for both service placement and migration are as follows. When a new service is requested in the edge, the first step is to select a sub-pool of resources (subtree), such that only the branches of the tree that contain the virtual resources available at the location of the user are considered. The initial service placement problem can then be described using a multi-objective optimisation function that finds the optimum server on the lowest subtree level possible. This approach thus provides a solution that will produce the minimum possible latency between the user and the services, and optimal usage of the resources based on the provider's goal, such as load balancing or energy efficiency.

However, as already discussed, the optimal placement will be optimal only as long as the vehicle remains in the service area of the base station where the service is initially requested. Once the vehicle moves to a different service area, the edge service must follow, so that the optimal latency principle is maintained. For these purposes, on each handover event, the resource management system analyses the latency parameter and, if necessary, activates a service migration event. The analysis is done by comparing the location of the service with the new location of the vehicle within the resource tree. If these two belong to different branches of the tree, then the service is migrated to the new tree branch that corresponds to the new vehicle location. Again, the exact edge server, where the service will be migrated, is determined based on a new multi-objective function that optimises the virtual resources use according to the provider's goals. In the cases when there is no possibility to migrate the service due to the lack of resources in the destination sub-tree leaf, the migration is aborted and the service runs in sub-optimal conditions.

For a more detailed description of the dynamic resource management system implementation in CloudSim, please refer to [9].

III. CASE STUDY: TOURISTIC INFOTAINMENT SERVICES

For the purposes of analysing the performances of the proposed dynamic resource management techniques for hosting vehicular networks services, a realistic urban case study scenario is developed.

The case study analyses the edge computing platform performance when hosting infotainment vehicular services providing information for tourists in the city centre of Alicante, Spain (Fig. 2). The city centre size is roughly 1.8 x 2.0 km and is covered using 9 5G base stations each with 200 m radio range. In Fig. 2, the location and service area of each base station are aligned with the city street map.

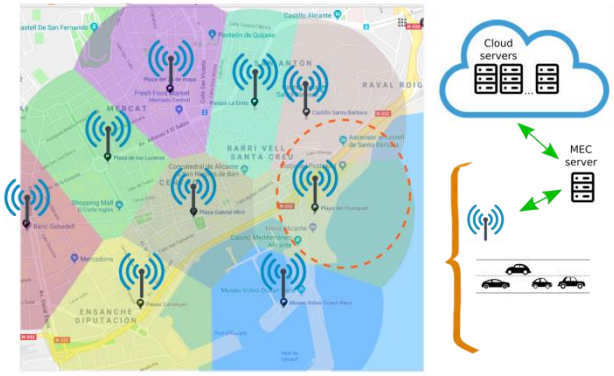

Fig. 2. Case study simulation area: the city centre of Alicante, Spain, using the original OpenStreet maps with 5G base stations co-located with famous touristic attractions and their corresponding service areas.

Each vehicle in the simulation is equipped with an infotainment system that requests a service at the moment when it enters the city centre and establishes a connection with one of the base stations. As the vehicle moves through the city centre, the infotainment services follow the mobility pattern of the vehicle with the dynamic resource management system migrating the service on the edge servers of the new base station that the vehicle uses. Once the vehicle leaves the city centre and is no longer associated with any of the base stations, the service is deallocated from the edge computing platform and the resources are freed.

In order to be able to create the defined simulation scenario, two simulation tools are integrated: CloudSim [10] for simulating the edge platform and the dynamic resource management system and SUMO [11] for simulating the environment using OpenStreetMaps, and the vehicular movements in the environment.

The city centre of Alicante represented OpenStreetMaps are used as input to SUMO, wherein the average vehicles per second is defined. The SUMO simulator generates the vehicles at the edges of the map and chooses a random route for the vehicle throughout the provided map. During the mobility of the vehicle, all traffic rules are embedded in the OpenStreetMaps, such as one-way streets, roundabouts rules, traffic lights, and speed limits in different areas (overall max of 50 km/h) (Fig. 3). Using the average vehicles per second value, multiple scenarios can be created: from light traffic examples to traffic jams.

An integration tool that uses the output from the SUMO simulator as an input to the CloudSim simulator is developed. The CloudSim simulator hosts the virtual network that interconnects the edge servers and manages the virtual resources. In addition, CloudSim is extended with the new placement and migration policies that implement the proposed dynamic resource management system, as well as the possibility to create and destroy services on the fly.

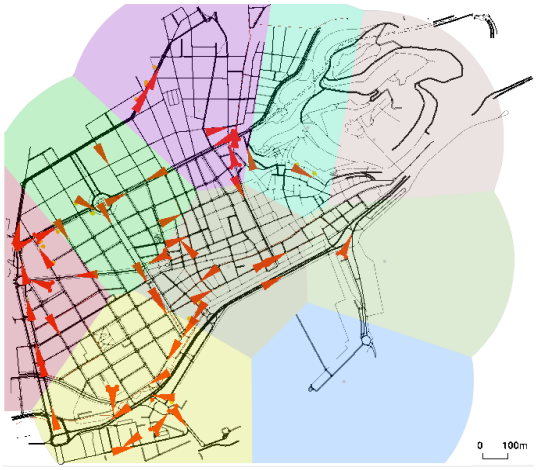

Fig. 3. Example of snapshot of the SUMO simulation. A small number of vehicles (triangles) moving in the simulation area based on the traffic rules.

As each vehicle is created in SUMO, a corresponding edge service is defined in CloudSim and the proposed initial placement policy is called to place the service in the edge platform. As the vehicle moves in the simulation area, SUMO sends information to the integration tool that tracks the vehicles mobility relative to the locations of the base stations using the Voronoi principle to decide when a handover event occurs. Each handover event is forwarded to CloudSim and the proposed migration policy is activated to ensure the corresponding service "follow-me" behaviour.

IV. CASE STUDY SIMULATION RESULTS

For the purposes of the performance analysis, a set of simulation scenarios has been created by varying the number of generated vehicles (0.5, 1, and 1.5 vehicles per second) that correspond to around 5700, 8600 and 12000 services, respectively, and the number of edge servers in the network (45, 63, 81, 99, and 117 hosts or 5, 7, 9, 11, and 13 hosts per base station). The three scenarios when varying the number of vehicles present three different traffic stages: light traffic, normal traffic with minimum congestion, and heavy congestion with multiple traffic jams. Each simulation is run for 3 h simulation time. For the multi-objective optimisation function, load balancing is chosen. Each scenario is run multiple times and the presented results are averaged across the obtained results.

In Fig. 4, the efficiency of the initial allocation of services is represented for a different number of hosts and services in the network. It is evident that the proposed algorithm works very well providing 100 % optimal placement (in the corresponding leaf of the tree) in most of the cases. Only when the service demand is very high compared to the available edge resources, the initial placement is not optimal and drops to 88 % for the worst-case scenario of only 45 edge servers and 12000 services.

In Fig. 5, the corresponding migration efficiency is presented when using the proposed technique. Similar conclusions can be drawn with the migration events being optimal in most of the scenarios.

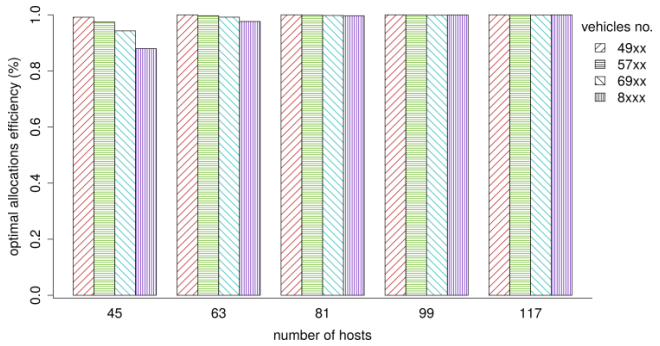

Fig. 4. Placement efficiency – percentage of optimal initial allocations for simulation scenarios with a different number of edge hosts and vehicles.

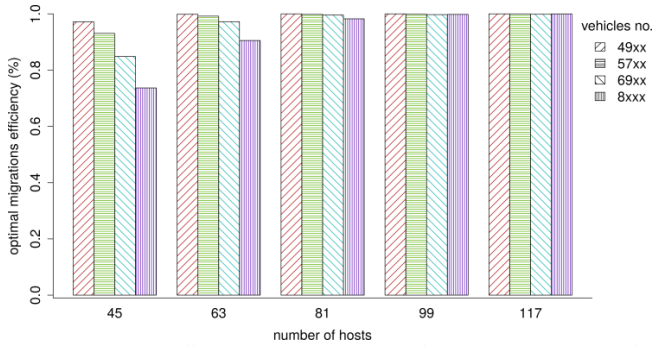

Fig. 5. Migration efficiency – percentage of optimal migrations for simulation scenarios with a different number of edge hosts and vehicles.

However, due to the higher sensitivity to available resources during the migration process, as one service uses double resources (at the old host and at the new hosts where it is migrated), optimality in migration is harder to achieve in the case of smaller number of hosts and higher number of services. The presented results can be used as input also when designing the system, providing insight into the minimum number of hosts needed to assure optimum performances even in the cases of traffic jam (12000 services).

When analysing the presented results considering the migration efficiency one needs to take into consideration that during the simulation there is a very large number of triggered migration events. For example, for the scenarios with around 8600 vehicles, the total number of migrations during 3 h simulation time is about 26000. Over 98 % of these migrations are optimally completed in the cases when there are at least 81 hosts in the provider network (Fig. 6).

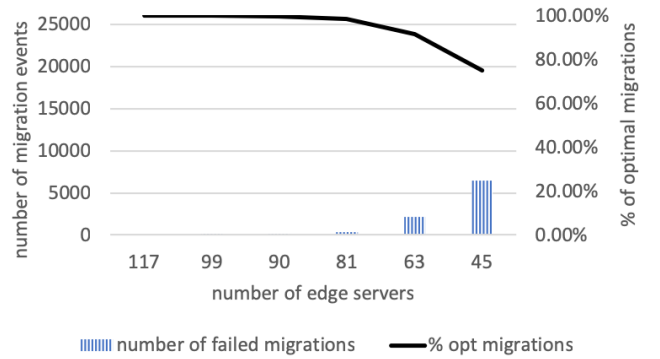

Fig. 6. Migration efficiency in details for the simulation scenario with around 8600 vehicles and about 26000 scheduled migration events.

In the worst-case scenario, there are around 6500 unsuccessful migration events leading to an efficiency of about 75 % when using only 5 edge servers per base station, each equipped with 6 cores, 12 GB RAM and 1 Gb network bandwidth.

V. CONCLUSIONS

In this paper, a dynamic resource management system for optimal usage of the available virtual resources in the edge network is proposed The proposed system implements techniques for optimal initial allocation and continuous migration of vehicular services, so that the lowest latency is provided throughout the service lifetime.

Using a case study for an infotainment vehicular services that can be used as added value service for tourists, the performance and efficiency of the proposed resource management system are evaluated. The simulation results show that the proposed techniques ensure the most optimal placement and migration whenever there are enough resources available to support the demand. Even in highly dense urban traffic scenarios, the efficiency does not drop below 88 % for allocation and 75 % for migration requests.

The presented results can also be used as during the system design stage, so that the base stations are equipped with the optimal number of edge servers in order to serve the projected number of service requests.

REFERENCES

[1] Z. Su, Y. Hui, and Q. Yang, "The next generation vehicular networks: A content-centric framework", *IEEE Wireless Communications*, 2017, vol. 24, no. 1, pp. 60–66. DOI: 10.1109/MWC.2017.1600195WC.

[2] V. W. S. Wong, R. Schober, D. W. K. Ng, and L.-Ch. Wang, *Key technologies for 5G wireless systems*. Cambridge university press, 2017. DOI: 10.1017/9781316771655.001.

[3] S. Kekki, W. Featherstone, Y. Fang, P. Kuure, A. Li, A. Ranjan, D. Purkayastha, F. Jiangping, D. Frydman, G. Verin, K.-W. Wen, K. Kim, R. Arora, A. Odgers, L. M. Contreras, and S. Scarpina, "MEC in 5G networks", *ETSI White* Paper, no. 28, 2018.

[4] D. Sabella, H. Moustafa, P. Kuure, S. Kekki, Z. Zhou, A. Li, Ch. Thein, E. Fischer, I. Vukovic, J. Cardillo, V. Young, S. J. Tan, V. Park, M. Vanderveen, S. Runeson, and S. Sorrentino, "Toward fully connected vehicles: Edge computing for advanced automotive communications", *5GAA Automotive Association White Paper*, 2017.

[5] T. X. Tran, A. Hajisami, P. Pandey, and D. Pompili, "Collaborative mobile edge computing in 5G networks: New paradigms, scenarios, and challenges", *IEEE Communications Magazine*, vol. 55, no. 4, pp. 54–61, 2017. DOI: 10.1109/MCOM.2017.1600863.

[6] R.-A. Cherrueau, A. Lebre, D. Pertin, F. Wuhib, and J. M. Soares, "Edge computing resource management system: A critical building block! Initiating the debate via OpenStack", *The USENIX Workshop on Hot Topics in Edge Computing (HotEdge'18)*, 2018.

[7] B. Blanco, J. O. Fajardo, *et al*., "Technology pillars in the architecture of future 5G mobile networks: NFV, MEC and SDN", *Computer Standards & Interfaces*, vol. 54, pp. 216–228, 2017. DOI: 10.1016/j.csi.2016.12.007.

[8] S. Fortunato and D. Hric. "Community detection in networks: A user guide", *Physics Reports*, vol. 659, pp. 1–44, 2016. DOI: 10.1016/j.physrep.2016.09.002.

[9] K. Gilly, S. Filiposka, and A. Mishev, "Supporting location transparent services in a mobile edge computing environment", *Advances in Electrical and Computer Engineering*, vol. 18, no. 4, pp. 11–22, 2018. DOI: 10.4316/AECE.2018.04002.

[10] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. De Rose, and R. Buyya, "CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms", *Software: Practice and Experience (SPE)*, vol. 41, no. 1, pp. 23–50, 2011. DOI: 10.1002/spe.995.

[11] D. Krajzewicz, "Traffic simulation with SUMO–simulation of urban mobility", *in Fundamentals of traffic simulation*, pp. 269–293, Springer, New York, NY, 2010. DOI: 10.1007/978-1-4419-6142-6_7.