



Budapest University of Technology and Economics  
Faculty of Electrical Engineering and Informatics  
Department of Telecommunications and Media Informatics

# Simulating and optimizing federated networks in a vehicular setting

**Scientific Students' Association Report**

Author:

Márk Konrád

Marcell Szabó

Advisor:

dr. László Toka

István Pelle

2022

# Contents

<b>Abstract</b>	<b>i</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Use case description</b>	<b>4</b>
<b>3 Related work</b>	<b>6</b>
3.1 Edge-focused Kubernetes platforms . . . . .	6
3.2 Federated learning in vehicular networks . . . . .	7
3.3 Distributing HD maps in vehicular networks . . . . .	7
3.4 D/M/1 queues . . . . .	8
<b>4 System model</b>	<b>9</b>
4.1 The choice of D/M/1 queues . . . . .	9
4.2 Theory for calculating the value of information . . . . .	10
4.3 Possible optimization avenues . . . . .	12
<b>5 Numerical simulation</b>	<b>15</b>
5.1 Testbed description . . . . .	15
5.2 Proof-of-concept application . . . . .	16
5.3 Performance evaluation results . . . . .	17
<b>6 Conclusion</b>	<b>24</b>
<b>List of Figures</b>	<b>25</b>

**List of Tables**

**26**

**Bibliography**

**26**

# Abstract

At the dawn of autonomous driving, vehicular communications and coordination become more vital than ever. Fast information gathering, processing and sharing creates the basis for safety and efficiency, the main promises of conceding the control of vehicles from humans to machines. In this paper we propose to deploy an information gathering and distributing system that aims exactly at minimizing the latency of delivering the essential information to the end clients. We specifically tackle the crowd-sourced maintenance of high definition maps, i.e., road maps with extremely high accuracy and environmental fidelity containing dynamic information about the traffic as well, via a federated analysis scheme, and by broadcasting those maps through a 5G network. The system is designed for minimizing the latency of information delivery: analytical models based on queuing theory and optimization are proposed, and a wide range of system parameters are evaluated in numerical simulations. We find that the latency of delivering timely high quality information to end clients can be reduced with careful dimensioning of the system. According to our measurements, high-speed 5G data connection is a must, as we reach the optimal latency by utilizing map segments with a 1 km diameter and Gb/s uplink speeds, in densely populated central metropolitan settings.

# Chapter 1

## Introduction

Since its first steps around two decades ago, through diverse virtualization options, cloud computing started to replace high-resource local compute equipment, so much so that now it weaves through our everyday lives. It backs some of the most mundane applications that we use most often accompanied by some artificial intelligence (AI) solution. While for some time the direction of moving compute tasks was clearly targeting the cloud, in recent years—due to latency or data location considerations—this trend has taken a slightly modified trajectory with the appearance of edge and fog computing. Software leveraging these concepts and respective devices can run closer to end-users and provide lower response times for increased user experience. The process is further fueled by the newer generations of mobile networks that provide edge computing resources that are already being leveraged by public cloud providers [2, 14, 22].

Although at a much slower pace, a similar direction is observable in the case of vehicular control tasks albeit incorporating the edge concept right from the beginning. While initially vehicles relied solely on onboard electronics for providing driving assistance, cooperation with roadside units (RSU) using vehicle to infrastructure (V2I) communication enabled them to extend their sensing and computing powers. As the increase in vehicular traffic infers more issues from traffic control, road safety, and environmental aspects these extensions are more than welcome. While the increase of onboard [4] and RSU [20] compute power is foretold for the near future, vehicle and cloud interactions are also getting deeper [4]. Such infrastructure paired with low-latency, high throughput network connections can enable sophisticated driving assist functionalities as well as improve autonomous driving capabilities.

This evolution points towards platforms and applications that provide built-in support for efficiently handling the innate hierarchy of the infrastructure resources rang-

ing from onboard devices through edge or RSU equipment up to centralized cloud data centers as a continuum. To leverage such environments for providing advanced vehicular support functions, we identify three goals to accomplish. First, our solution has to be prepared for handling high-mobility vehicular environments for managing connections between static roadside and moving vehicular units. In the infrastructure continuum, each hierarchy level provides different compute and latency characteristics. While onboard equipment can run calculations with minimal latency, its compute resources and energy consumption are severely limited. Leveraging low-latency wireless network connections, edge/RSU devices still provide fast reaction times with higher compute and energy supply options. Although (beyond) 5G networks aim to support high throughput with low latency, the amount of data transmitted via the wireless link is still crucial and needs to be minimized. The cloud can offer virtually limitless compute power with no energy restrictions with the drawback of having potentially high latency. Thus, our second goal is to take into account latency and resource footprint aspects as well. Our solution needs to perform data acquisition, aggregation, and processing on the most adequate level in the infrastructure hierarchy and provide options for sharing and exchanging source and derived data among participants quickly and using as few computational resources as possible. In a high-mobility environment, this also implies that we need to exploit data locality and process data near its origin and distribute processed derivatives only to where they are requested. The multitude of vehicular sensors and increased processing power throughout the infrastructure enable constant frequent updates on the behavior of transport participants and road conditions. This can be exploited using AI applications that provide support for human drivers in the form of assisted driving or be part of the control loop in autonomous driving. Consequently, our third goal is to support such assisted or autonomous driving scenarios preparing for AI applications.

We argue that these aspects can be satisfied by leveraging federated analytics (FA) [11] in a 5G edge network. FA is a novel practice of applying data science methods to the analysis of raw data that is stored locally on users' devices. It works by running local computations over each device's data, and only making the aggregated results—and never any data from a particular device—available to the central server that is assumed to be an edge compute node integrated in the 5G network in our envisioned scenario.

In our view, this concept maps well to a multi-tiered infrastructure where vehicle nodes can collect data, edge/RSU units can aggregate and distribute the insights from localized data without breaching data privacy and without the need of V2V

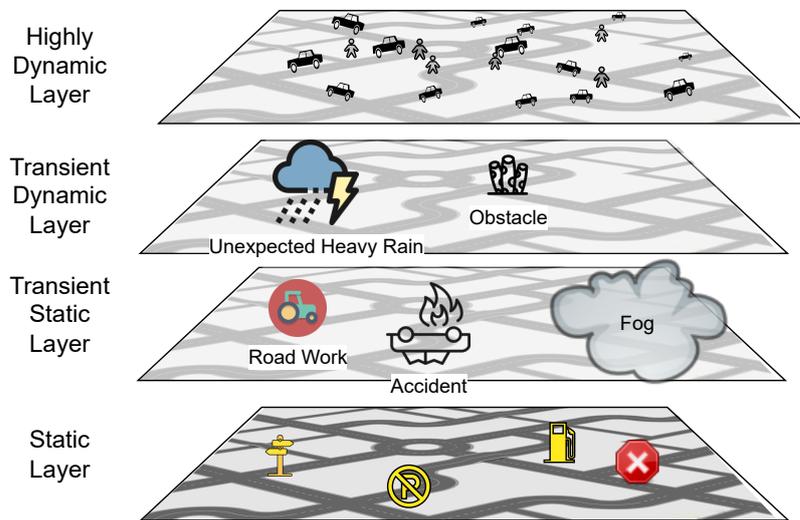
communication. FA also allows for reducing transmitted data sizes, as exchanging insights is sufficient in most cases instead of uploading raw data.

Consequently, our contributions are the following. We showcase an FA-based solution for collecting and aggregating data originating from vehicles, and we propose an analytical model that expresses the end-to-end latency of channeling information through the system. We also provide a numerical evaluation of our solution highlighting its prowess in low-latency information delivery in function of several parameters that must be defined in the phase of system dimensioning.

In order to discuss these aspects, the rest of our paper is organized according to the following structure. A deeper description of the envisioned system and service is given in Chapter 2. In Chapter 3 we review related work discussing information delivery solutions in vehicular networks, and queuing theory models that we build our analysis on. Then, we describe our system model in Chapter 4, detailing the FA-based upstream scheme. We highlight important aspects of our testbed, present the details of our test scenarios and provide their evaluation in Chapter 5. Finally, in Chapter 6, we summarize our work and draw the conclusions.

# Chapter 2

## Use case description



**Figure 2.1:** Representation of HD map consisting of different layers with static and dynamic information.

With the gradual shift towards autonomous driving, digital maps must be built for self-driving vehicles and go beyond navigation assistance. Autonomous Vehicles (AV) require high-fidelity information about the surrounding environment with centimeter-level precision to make crucial decisions. Numerous sensors are placed on the vehicles to collect the surrounding environment data needed for autonomous deriving, and AVs process those in real-time to make decisions. However, achieving autonomous navigation in real-time is challenging, especially in urban scenarios, due to the limited range of sensors and inherited inaccuracies. High Definition (HD) maps help autonomous vehicles perceive the surrounding environment and precise localization [17], which enables them to perceive beyond sensor visibility, apply context awareness of their environment, and process local road rules to make safer decisions and plan proactively.

HD maps, mainly built for self-driving purposes, have a high level of accuracy as vehicles need precise localization and environment data to maneuver in real-time. Initially, HD maps are created using special vehicles equipped with very precise sensor equipment, like Differential GPS (D-GPS), a multitude of cameras, and highly accurate laser scanners to collect the present obstacles and the traffic rules which apply to the surrounding environment. As shown in Figure 2.1, HD maps consist of different layers. The static layer contains information that is changed infrequently, such as a road map, the transient static layer shows information that remains the same for a time period, i.e., road work, the transient dynamic layer provides frequently changing information, and highly dynamic layers feed the real-time data of surrounding environment to the vehicle such as vulnerable road users. Since the road network is not a static environment and is constantly affected by changes, for example, introduced through traffic jams, accidents, construction works, or the current status of adaptive traffic signs, the map has to be continuously updated to provide the car with up-to-date information. It is quintessential to receive an updated HD map in very low latency to facilitate autonomous driving. Edge computing presents a solution to reduce network latency significantly. We make the case of deterministic update periods, and in subsequent sections we evaluate the toll that short update periods take on the system.

HD maps are location specific and require the transmission of large data volumes. The vehicles in the same geographic location request the same HD maps data (which we call a tile) for autonomous driving. This leads to the repetitive transmission of large volume HD map data over the core network, which stresses the capacity-limited backhaul links. Processing and storing the HD maps on Multi-access Edge Computing (MEC) servers at the vehicular network edge (RSUs) can alleviate the backhaul load and substantially reduce the latency, which is pivotal for autonomous driving. Thereby, vehicles can obtain the required HD maps from their connecting RSUs via vehicle-to-infrastructure (V2I) communication without going through the core network, or better yet, 5G edge can be leveraged as suitable infrastructure, providing broadcast communication as of Release 16 [7] and the necessary computing capabilities for FA in the vicinity of the users.

# Chapter 3

## Related work

### 3.1 Edge-focused Kubernetes platforms

Targeting the edge cloud calls for a suitable deployment platform. While the base Kubernetes distribution is adequate for handling the tasks of setting up multiple components at different points in a cluster consisting of multiple nodes, others exist to provide better performance in an edge environment. As this issue has effects on a growing number of application domains, so has it become widely researched by the community and the industry as well. Presently, several production-ready solutions have been developed and adopted, which are then further reviewed for their capabilities. [15] compares the performance of serverless use-cases of the most popular edge-centered Kubernetes distributions, K3s and MikroK8s, to the full-fledged Kubernetes. These distributions eliminate components that offer little, to no advantage for an edge-cloud environment, but they retain the core functionality of container orchestration. As a result, the deployment process is eased, and more importantly, the hardware requirements are alleviated. Specifically, K3s and MikroK8s require 512 MB and 540 MB of memory respectively, with both needing just 1 CPU core for operation.

In terms of performance testing, [15] proposed a complex methodology of using microbenchmarks with real-world workloads applicable to edge computing scenarios. Tests targeted distinct aspects of overall serverless performance, ranging from CPU and memory utilization, through the speed of I/O operations, to networking capacities. Thorough testing showed there is minimal difference between the two edge distributions in almost every case. In test cases where cold startup latency, disk throughput, or some CPU-bound tasks, like matrix multiplication and linear equations solving, were tested, they decisively outperformed the traditional Kubernetes

platform. On the other hand, the full Kubernetes system has a clear advantage in test cases where a heavy load is managed by only a single replica, and no scaling is allowed. If scaling is also permitted, then MikroK8s performs the best, although K3s is still close in every metric.

## 3.2 Federated learning in vehicular networks

This work focuses on the concept of multi-layer, universal federated learning frameworks in V2X applications. There are several industry-leading solutions already on the market, but none of them could facilitate our use case. In 2022 the most used open-source federated learning frameworks are FATE [18], Flower [3], and Tensorflow Federated [1]. After experimenting with these solutions, we found that none of the systems can be architecturally customized enough to develop our universal, multi-layered solution since they mostly focus on creating machine learning algorithms for federated learning. Work in [23] introduces a novel Federated Vehicular Network architecture in which it assigns a manager to a certain group of vehicles that would act as a proxy between the worker vehicle and the cloud server. The manager aggregates the worker models and distributes the model updates via Federated Vehicular Cloud. However, this work assumes that the vehicles are stationary and hence cannot be applied for high mobility scenarios.

## 3.3 Distributing HD maps in vehicular networks

Existing works on content distribution in vehicular networks only consider static content and fail to consider the dynamic information change in the HD maps. The HD map distribution problem differs from normal content distribution due to frequent changes in HD maps data that must be disseminated to all caching locations and end clients. Therefore distributing the HD map generates periodic traffic not only in access networks, but on the backhaul connecting to the RSUs too.

The authors of [26] focus on energy-efficient map data distribution. They propose an algorithm that minimizes the power consumption while receiving HD maps. An RSU serves a vehicle only if the energy required to receive data from the RSU and for basic movement is less than the remaining energy of the vehicle. For such vehicles, the data is divided among all the RSUs in proportion to the received power to provide the service. The authors in [25] propose a joint power control and spectrum assignment policy to maximize the sum data rate of the overall network for HD map

dissemination. They study the interference effect on data transmission and formulate a model that describes the interference control problem in V2X-enabled HD map dissemination. The authors suggest cooperative delivery of HD maps through V2I and V2V communications by dividing the HD map into many data blocks based on data volume and infrastructure environment. The work in [27] discusses HD map caching for autonomous driving in vehicular networks with unknown vehicle requests and trajectories. The proposed caching algorithm defines a reward function that considers the historical tile request data.

A new architecture is proposed in [21] that combines Multi-access Edge Computing (MEC), and Software Defined Networking (SDN) to enable HD map-assisted autonomous driving. A two-tier server structure is proposed with cloud and MEC servers to achieve resource utilization and network scalability. The applications and services are deployed on the MEC server using Network Function Virtualization (NFV) at the network edge. A MEC system framework is proposed in [28] for HD map applications. The authors introduce the application mode, functional modules, HD map data distribution workflow, and communication process between the autonomous vehicle client and the server.

### 3.4 D/M/1 queues

In order to model our federated HD map distributor network, we needed a queuing system which can handle fixed, periodical arrivals and Markovian service times. The only system models capable of handling these constraints are D/M/1 queues.

The authors in [6] describe an analysis of the D/M/1 queue with deterministic customer impatience, however, the approximations are not suitable for calculating the waiting time of each customer after arrival. The study in [13] goes into great details about D/M/1 systems. The author carefully analyses every aspect of the network, from costs, probabilities to idle and waiting times. The theory described in this paper was essential for creating the calculations discussed in Chapter 4.

# Chapter 4

## System model

In this section, we propose a mathematical model based on D/M/1 queuing systems that can describe our federated HD map distribution network and a method to measure the value of information transmitted between clients and base stations. After the model description, with the help of these formulas and the necessary constraints, we analyze the output of known numeric optimization algorithms in order to formulate some guidelines and best practices for creating federated networks with optimized performance.

### 4.1 The choice of D/M/1 queues

We chose D/M/1 queues to model our federated network. In this type of queue, arrivals occur periodically in a deterministically fixed  $\beta$  (denoted by  $T_t$  in later discussions) time and service times are exponentially distributed. The reason behind this choice is that neither M/M/1 nor M/D/1 queues are suitable to depict the importance of fixed periodical trigger times in each HD map layer. Since the queues have fixed arrival times, we can only account for latency and other uncertain factors at the service time part of the model which, like in many other networking based queuing systems, assumes exponential probability distribution for service times. This exponential distribution is suitable for our use-case, based on the assumption that the service times in our model cover the delay that the construction and upload of insights from a number of clients produce, in addition to the FA aggregation itself in the edge node and the down-link communication time.

As mentioned in Chapter 2, the HD map is built by using a number of different layers with different trigger periods based on the dynamicity of information depicted by them. For each HD map layer, we have an independent closed D/M/1 queue that

accounts for both the uplink and downlink communication in their corresponding tile.

## 4.2 Theory for calculating the value of information

The calculation of the  $T_w$  waiting time of an HD map layer  $i$  in such queues is essential for determining the value of information transmitted throughout the network. As per our model and the analysis of D/M/1 queues presented in [13],  $T_w$  can be obtained as  $\delta/((1 - \delta)\mu)$  with  $\delta$  being the smallest absolute root of Eq. 4.1:

$$\Delta = e^{-\mu T_t(1-\Delta)} \quad (4.1)$$

where  $\Delta$  is an internal parameter used to calculate waiting time, variance and other attributes of D/M/1 queues in accordance with [13]. This root can be calculated using the Lambert  $W$  [5] function (see Eq. 4.2), and consequently, we can get the  $T_w$  waiting time of each layer as per Eq. 4.3.

$$\delta = -\frac{W(-e^{-\mu T_t} \mu T_t)}{\mu T_t} \quad (4.2)$$

$$T_w = \frac{\delta}{(1 - \delta)\mu} = \frac{W(-e^{-\mu T_t} \mu T_t)}{\mu^2 T_t (1 + \frac{W(-e^{-\mu T_t} \mu T_t)}{T_t})} \quad (4.3)$$

where  $W$  is the Lambert  $W$  function.

To limit the possible space of parameters, we introduce the constraints defined in Eq. 4.4 (see Table 4.1 for the description of our notations, where each symbol represents a parameter of a single queue). Constraint ( $C_1$ ) ensures that there is an upper limit on the number of clients dictated by the size of transmitted data, trigger time and uplink capacity. Since preferably all clients should get back the aggregated information, i.e., the up-to-date local HD map, constraint ( $C_2$ ) makes sure the waiting time is less than the trigger time. Additionally, constraint ( $C_3$ ) is required to ensure the waiting time function returns with a value in the domain of real numbers and it obeys the constraints of D/M/1 queues mentioned in [13]. Otherwise this value can be in the complex domain because of the utilized Lambert  $W$  function. The constraints ( $C_4$ ), ( $C_5$ ) and ( $C_6$ ) keep the service rate, trigger time and the number of clients within realistic bounds, as it is unrealistic to have a service rate and trigger

**Table 4.1:** Mathematical notations used in our system model.

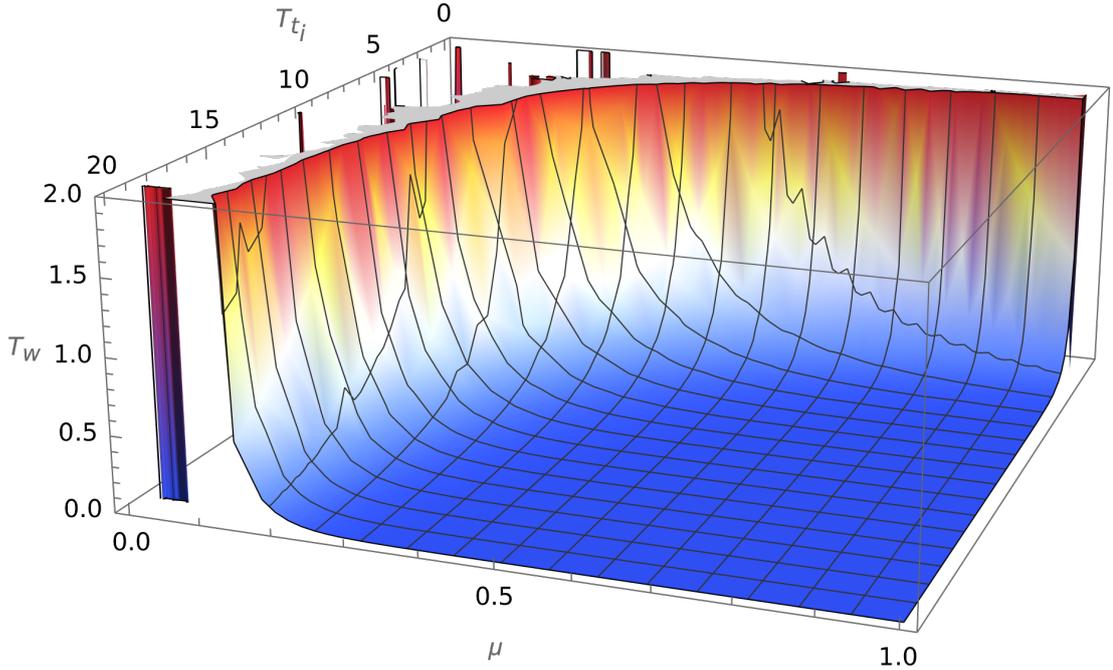
Notation	Description
$N_c$	Number of clients
$N_d$	Amount of data uploaded by each client
$B$	Total bandwidth allocated to the HD map layer
$T_t$	Trigger time of the HD map layer
$T_w$	Waiting time calculated for the HD map layer
$T_{p^j}$	Preparation time for vehicle $j$
$T_{r^j}$	Response time vehicle $j$
$\mu$	Service rate for the queue representing the HD map layer pipeline
$I$	Information value of an HD map tile
$W$	Lambert $W$ function

time under 0 and the client amount under 1. Figure 4.1 shows, within these bounds, the change of the waiting time ( $T_w$ ) based on the service rate and the arrival times.

$$\begin{aligned}
 (C_1) : \frac{N_c N_d}{T_t} &\leq B \\
 (C_2) : T_w &\leq T_t \\
 (C_3) : T_t \mu &> 1 \\
 (C_4) : \mu &> 0 \\
 (C_5) : T_t &> 0 \\
 (C_6) : N_c &> 1
 \end{aligned} \tag{4.4}$$

As shown in Eq. 4.3, the waiting time is inversely proportional to the trigger time, therefore the logical optimization for the trigger time is to minimize them, while obeying to the constraints in Eq. 4.4. In Chapter 5 our experiments apply trigger times no smaller than 1 s. This is because our use-case does not require sub-second trigger times to achieve a reasonable information density.

$$\begin{aligned}
 I_1 &= 4 \frac{N_c}{\sqrt{600 + N_c^2}} \\
 I_2 &= \begin{cases} \frac{1}{T_w - 1 + T_t} & \text{if } T_w \leq 1 \\ \frac{1}{\ln(T_w + T_t)} & \text{if } T_w > 1 \end{cases} \\
 I &= \frac{\arctan(I_1 I_2)}{\frac{3\pi}{7}}
 \end{aligned} \tag{4.5}$$

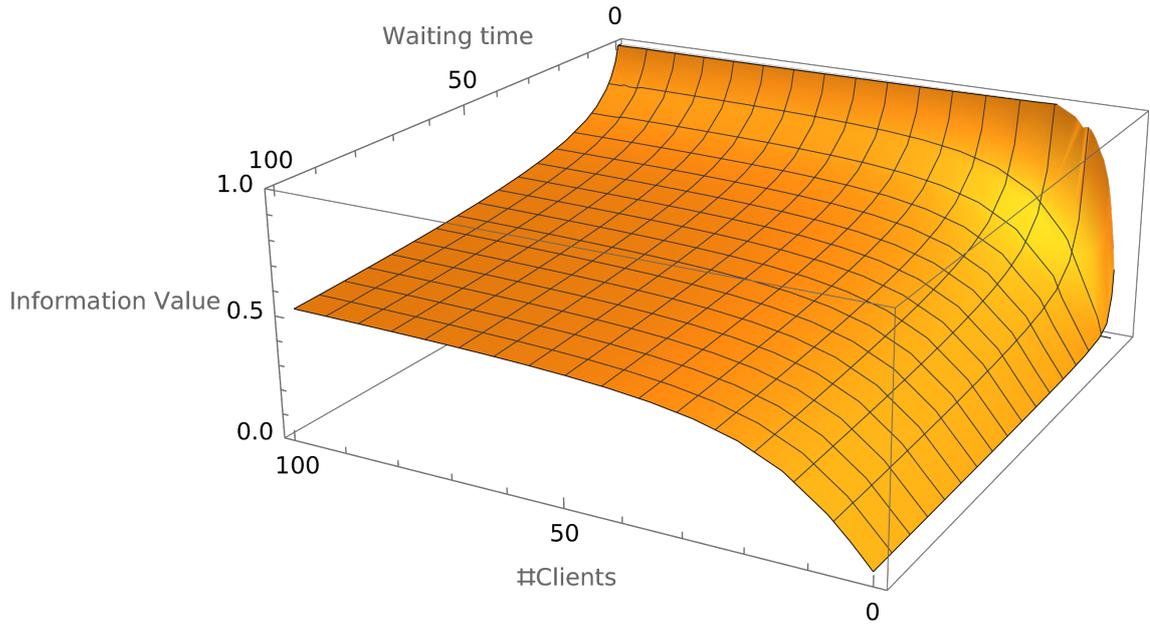


**Figure 4.1:** The change of  $T_w$  based on network properties

To measure the value of information transmitted within the network, we propose Eq. 4.5. To calculate the  $I$  information value, we chose two functions to represent the two main moving parts of the system. The first function ( $I_1$ ) is for considering the number of clients in the geographic area in focus, and the second function ( $I_2$ ) is to consider the waiting time needed to get the most recent information from the base stations, also taking into account the trigger time. The number of communicating clients is important because if there are only a handful of devices present in the network then the aggregated data contains less information about the environment, and after a certain number of clients the transmitted information becomes highly redundant. The chosen sigmoid-like function is perfect to depict this phenomenon. The metric, based on the waiting and trigger times, is essential to measure the information recency. To make sure the information value function falls within an easily interpretable 0 to 1 range, we used arctan and some carefully chosen constants for a scaling transformation.

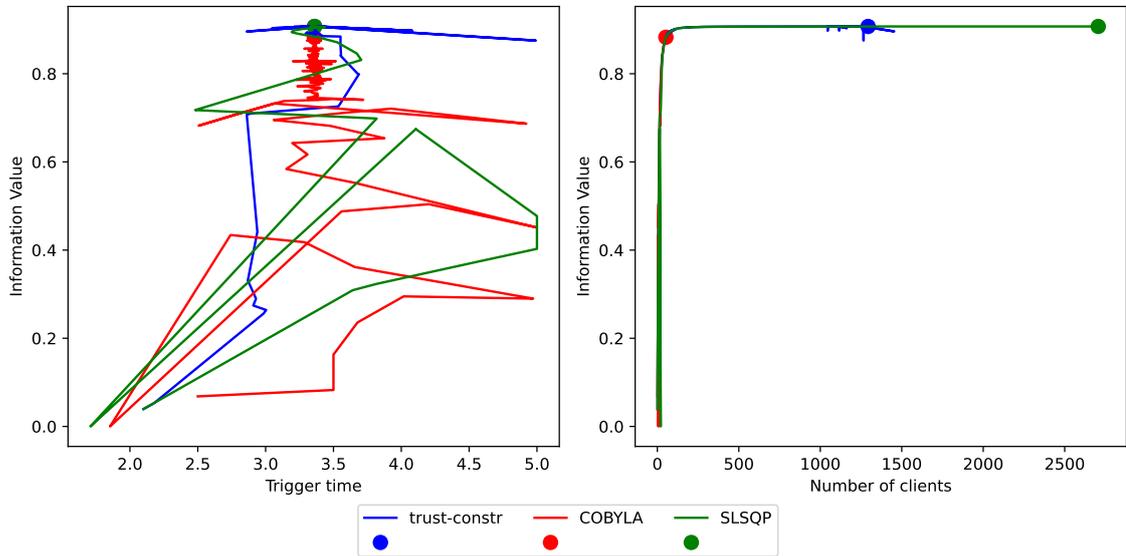
### 4.3 Possible optimization avenues

As the proposed  $I$  and  $T_w$  functions proved to be too complex for generating meaningful results with analytical optimization techniques, we chose three different numerical optimization methods to reach the desired results. We leverage the Trust Region Constrained Algorithm [24], COBYLA [10] and SLSQP [16]. The paths



**Figure 4.2:** The change of  $I$  based on  $T_w$  and  $N_c$

these algorithms took to optimize the information value can be seen in Figure 4.3 where the dots mark the end results each of them reached.



**Figure 4.3:** Different parameters after optimization

The optimization was performed over two input variables: the trigger time of a layer, and the number of clients. The  $\mu$  parameter of Eq. 4.3 was set to 0.5 and in the experiment depicted in Figure 4.3, the bandwidth was 1 Gb/s with the transferred data size being 1 MB. The optimization was performed for a relatively dynamic HD

map layer, so an additional constraint was added to keep the trigger time under 5 seconds.

From the results of different optimization runs it became apparent, that the changes in the waiting times are almost negligible in the range needed for optimal HD map transfer because of the properties of D/M/1 queues, assuming instantaneous uplink and downlink transfers.

The measurements also gave us insights on how the relation of bandwidth and information value can lead to an optimal network. When the bandwidth drops, the number of serviceable clients drastically decreases too, but the value of information can be retained by dynamically decreasing the trigger time of the clients. This is especially important for highly dynamic HD map layers, where even a small number of clients can provide a reasonable amount of information if their trigger time is frequent.

# Chapter 5

## Numerical simulation

In order to reinforce our theoretical model for the proposed federated HD map aggregating network, we are introducing empirical evidence based on widespread simulation scenarios. First, we thoroughly discuss the high-level implementation of our testbed, then we proceed by defining the context and the metainformation for our simulation scenarios. When every core knowledge is shared, we introduce and explain our results. Lastly, we draw conclusions based on the received output and summarize them to provide exact answers about the practical applications of our system. We note that for the sake of simplicity we did not leverage an edge focused Kubernetes distribution, but every component can be adapted to one.

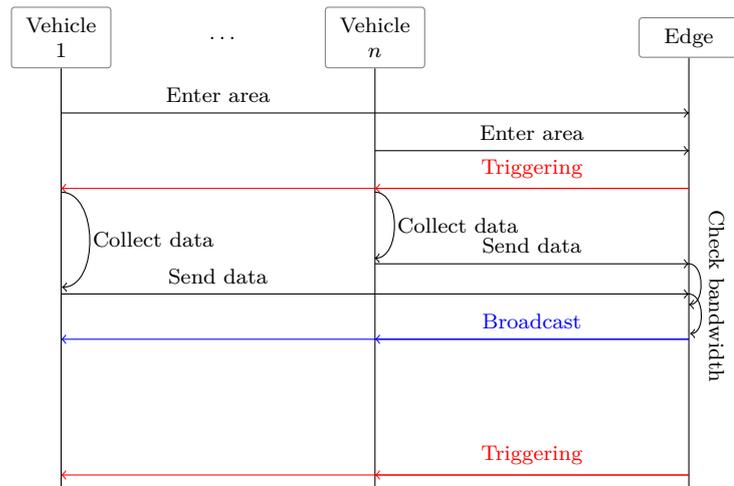
### 5.1 Testbed description

Our testbed consists of a custom-written application simulator that relies on the Simulation of Urban MObility (SUMO) framework [8]. It has become an industry leader solution for virtual vehicle simulation because it can easily handle large traffic networks, while holistically managing every prospective actor, ranging from automobiles and pedestrians to traffic lights and road-side equipment. Connecting this framework to our defined, application-level testbed was made possible with the use of the TraCI interface [9]. It provides an interface through which our simulator gained access to every volatile data regarding the test environment, most importantly to the relative position of the vehicles to the defined edge node.

## 5.2 Proof-of-concept application

On the implementation level, as the simulation in Sumo gradually unfolds, we perform the operations required for the federated HD-map analytics in the same fashion. Every member of each actor group has a step listener which executes data-driven communications between the actors when prompted by the TraCI interface. In our proof-of-concept application, we only consider vehicles and edge nodes, but these can be optionally expanded.

During the initialization of the Sumo simulation, we create a Python object for the edge node and get its step listener registered into the event cycle through the TraCI interface. This process is also performed for each vehicle that appears in the simulation throughout the entire course of the simulation period. After their registration, the step listeners get invoked at every discrete timeslot when a pre-defined step-length duration of time has passed, performing their actions periodically, based on the current status of their objects. The concrete proceedings can be followed in Figure 5.1.



**Figure 5.1:** System flow of federated analysis of HD map data

The edge node regularly triggers vehicles that are residing in its transmission area. In our terminology, this is called the trigger time ( $T_t$ ). Upon this trigger stimulus, the vehicles collect, package, and then send data regarding their immediate surroundings, to update a specific HD map layer. This period is labeled preparation time ( $T_p$ ). The vehicles then wait for each other to finish their data preparation phase, enabling the edge node to perform the data aggregation via FA. When this is complete, all vehicles which are present in the transmission area by this time, receive the aggregated information as the latest version of the respective HD map layer. For those vehicles which provided the data, this period is referred to as response time

( $T_r$ ). In other words, this physical quantity is the end-to-end latency in a traditional FA setup. Also, it is important to notice that  $T_p + T_r$  is the same among all training vehicles, equaling to the  $T_w$  waiting time defined in Chapter 4. Finally, after the aggregation, until the current trigger period lasts, all participants are idle, as they wait for the following trigger stimulus.

Note that here we described the system with a single edge node, although, in a practical example, multiple edge nodes may be placed to cover a larger area, e.g., cities or highways. Moreover, HD maps are known to have multiple layers, each containing information with varying volatility, so each edge node should be able to maintain every HD map aggregation separately. Our proposed system can incorporate these refined requirements simply by transmitting newly aggregated map layers between neighboring edge nodes and by defining distinct trigger periods for every HD map layer.

### 5.3 Performance evaluation results

Throughout our simulation scenarios, we conducted several test cases to determine the effects of various parameters. These are mostly included in our mathematical model description in Chapter 4, like the exponential distribution of  $T_p$ , the length of  $T_t$ , or the actual density of clients in the transmission area. This last parameter is defined by numerous factors, like the overall vehicle density in the input map, the diameter of the edge node transmission area, and also by the uplink bandwidth that is available for the clients. We summarize all the factors determining our results in Table 5.1.

We used random generated route patterns for vehicles, to produce the targeted average vehicle density. For this, studies [12, 19] suggest the order of magnitude of 100 vehicles/km<sup>2</sup>, thus we chose the density of 200 vehicles/km<sup>2</sup> in order to be aligned with the related work and have reasonable simulation runtimes. We set the data size to 300 kB by assuming privacy preserving FA insights being uploaded from each client.

Other attributes had their set of values defined on a logarithmic scale, as listed in the respective rows of Table 5.1. We took the Cartesian product of these value sets so that one particular result, e.g., a tuple of (500 m, 1 s, 10 Mb/s), dictated the input values for one simulation scenario. We performed every scenario 10 times and took the average of the results. In each case, the target variables were the information value which is the numerical output of the information function defined in Chapter 4, and the waiting time for the vehicles within the trigger period.

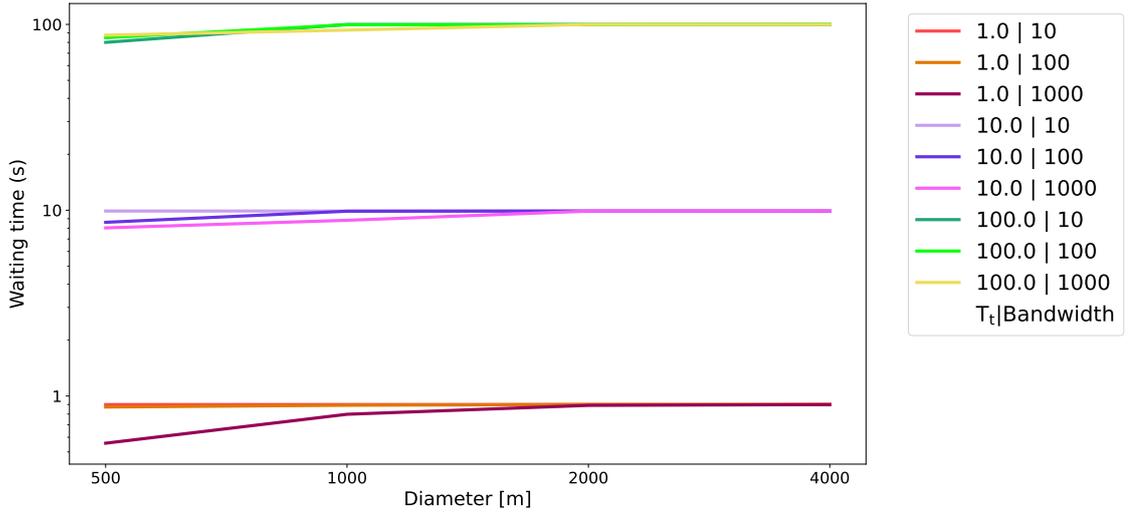
**Table 5.1:** Attributes and their investigated values in simulation scenarios

Attribute name	Set of values
Vehicle density	200 vehicles/km <sup>2</sup>
$\mathbb{E}[\text{data size/vehicle}]$	300 kB
$\mathbb{E}[T_p]$	$T_t \cdot 0.1$ s
Edge diameter	500, 1000, 2000, 4000 m
$T_t$	1, 10, 100 s
Uplink bandwidth	10, 100, 1000 Mb/s

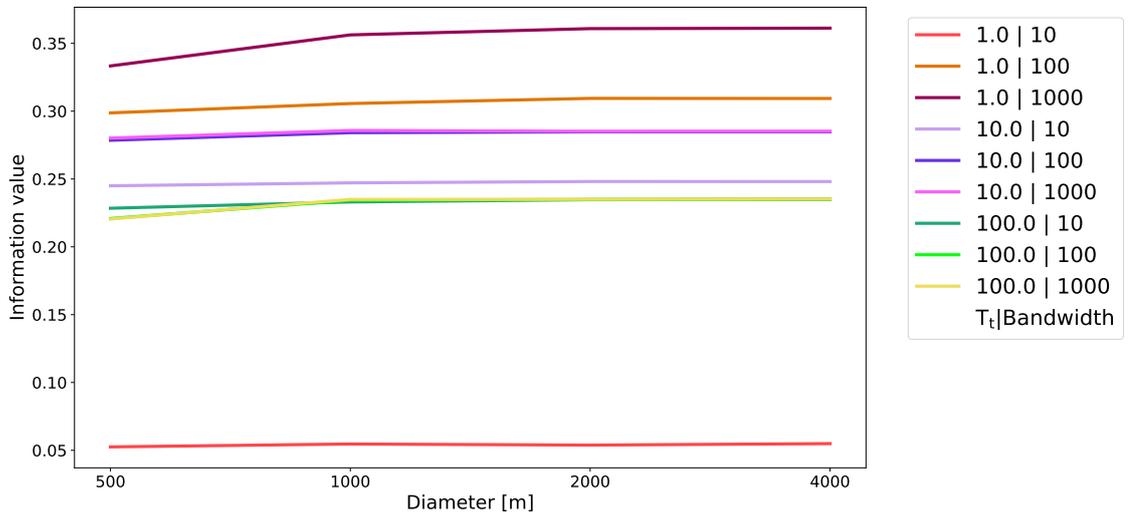
**Diameter:** In Figures 5.2 and 5.3, we depict how the waiting time and the information value react to the diameter change. At first let us observe Figure 5.2: in every test scenario the waiting time converges to the trigger time as the diameter grows, given that more vehicles become reachable, so the uplink load becomes greater up to the same bandwidth limit. This is more conspicuous if we check the lines with lower trigger time and high Gb/s uplink bandwidth because these approach the trigger time at a much slower rate as they can accommodate more traffic. On the other hand the impact of trigger time is almost equally tremendous. As the trigger time becomes larger, gradually the effect of uplink bandwidth becomes more indifferent towards the volume of waiting time. Figure 5.3 depicts the relation between the information value and the diameter: we can discover a similar phenomenon. The reason is again the growing vehicle density in the transmission area.

The optimal diameter can be found at around 1000 m in most cases if such dense areas are considered because the information value reaches its peak at this point while the waiting time does not deteriorate much compared to the 500 m diameter scenario. If the observed area has lower vehicle density, then a bigger diameter can provide better quality of service: low waiting time and high information value.

**Trigger time:** Figures 5.4 and 5.5 show the effect that the change of trigger time has on both the waiting time and the information value. In the case of the former (Figure 5.4), every test scenario is strikingly linear on the log-log scale. This is mainly due to the fixed link between the expected value of the preparation time and the trigger time in our simulation input (see Table 5.1). In terms of steepness, there are slight differences between the lines: when expanding the available uplink bandwidth, the clients get more chance in each trigger period to finish data collection before the new trigger period starts. Thus, they are more likely to attain a lower response time (latency), conclusively lowering the waiting time. This phenomenon is true until the vehicle density enables it. As soon as the area becomes overpopulated compared to what the uplink capacity can carry, the response time also starts to



**Figure 5.2:** Effect of the diameter on the waiting time



**Figure 5.3:** Effect of the diameter on the information value

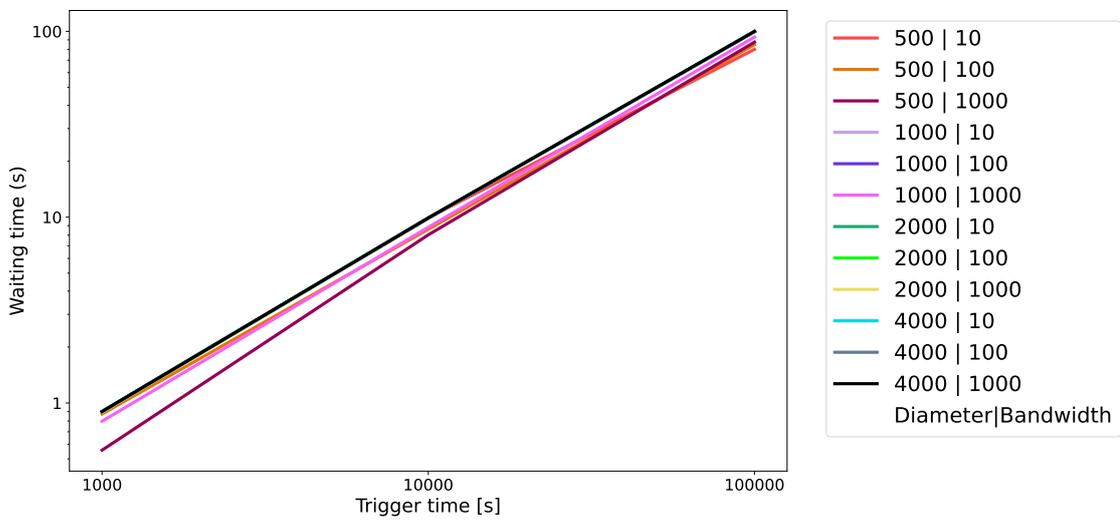
grow, since the vehicles are not able to finish their uploads soon enough. Hence, the respective lines become steeper in the figure.

When considering the information value in Figure 5.5, let us divide the plots with medium or high bandwidth from the ones with low bandwidth and examine them independently.

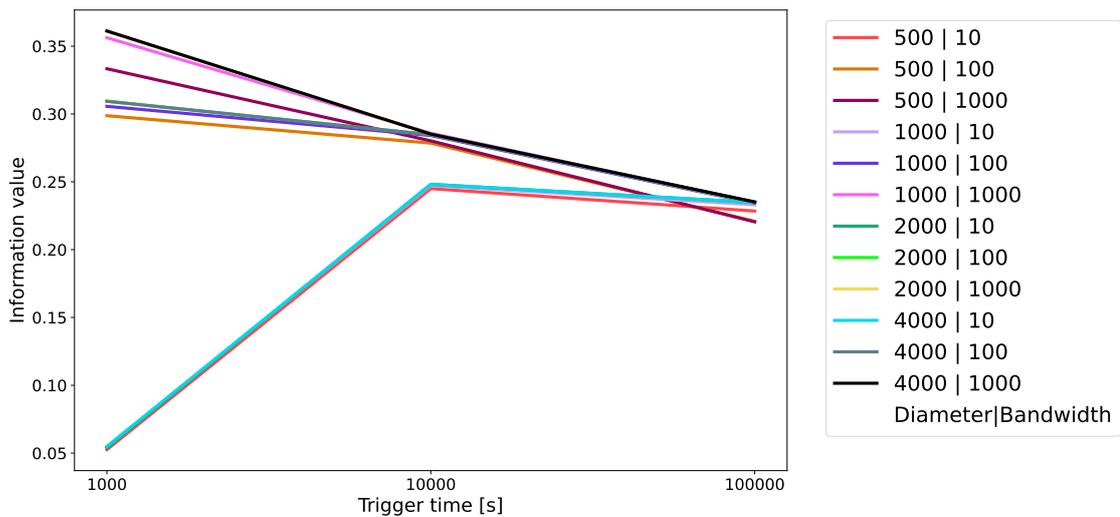
Upon observing the first group, it is important to notice these declines as the trigger period widens. However, the convexity/concavity of the function varies across the test scenarios. This feature is mainly dependent on the uplink bandwidth. If it is in the 100 Mb/s region, the function is concave, conversely, if it reaches the volume of Gb/s, the function becomes convex. Regardless of the shape, each function has its maximum at the beginning of the scale, at around 1000 ms. This is a direct cause of

the fact that we considered only urban areas with high vehicle density. When these circumstances are given, it is advisable to maintain a low trigger time because then the information value is at its highest. Other factors, on the other hand, such as the characteristics of a specific HD map layer, the incurred transmission pollution, or some cost-related insights should also be taken into account when setting the trigger time.

Moving on, to the second observation group, the features are highly distinctive. Every function stays concave, with their maximum value only reached around 10000 seconds. As a result, the optimal setting differs from the previously stated, so when the available bandwidth is low, it is advisable to make use of larger trigger times.

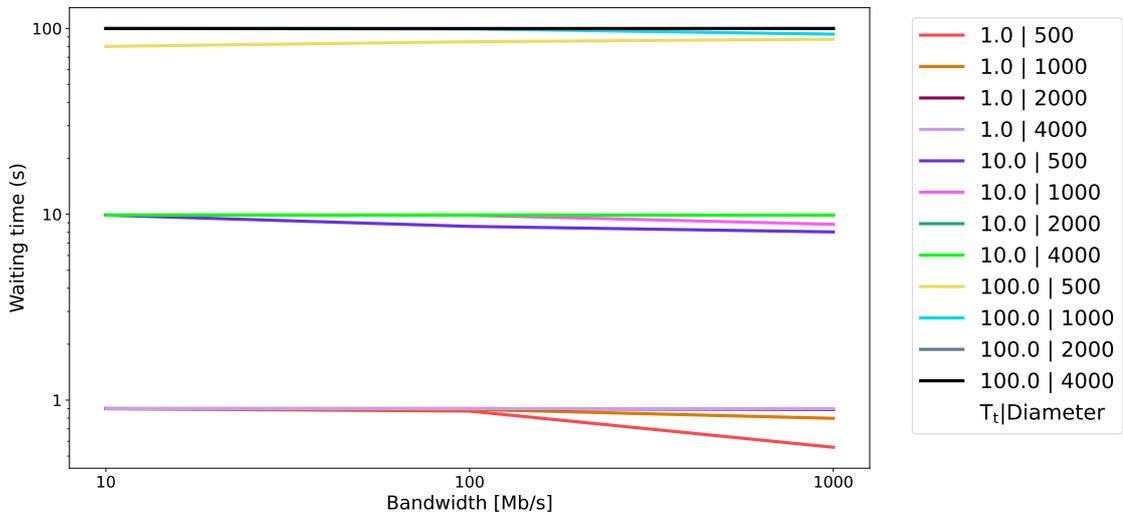


**Figure 5.4:** Effect of the trigger time on the waiting time

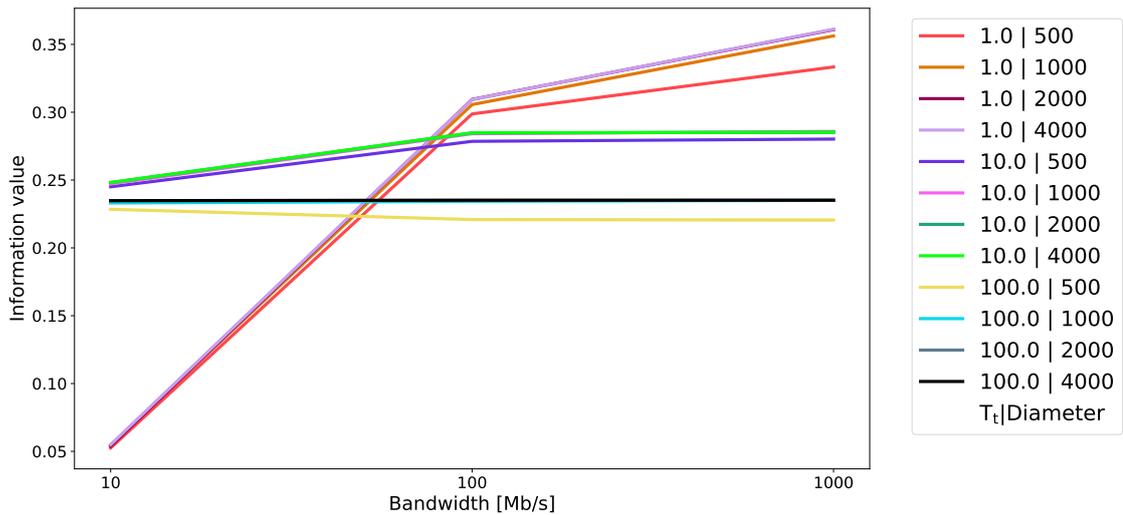


**Figure 5.5:** Effect of the trigger time on the information value

**Bandwidth:** Similarly to the prior plots, we depict the effect that the change in the available uplink bandwidth causes on the waiting time and information value. Figures 5.6 and 5.7 display some similarities to the diameter-related trends, especially in waiting time metrics. The uplink bandwidth has an immense factor on  $T_w$ , stemming from the fact that it is one of the two main contributors which determine whether vehicles can achieve a low  $T_p$ , or even finish, in a specific trigger period. As the available uplink bandwidth spreads out, so does the waiting time become increasingly lower across all scenarios, however more so in cases with a smaller diameter.



**Figure 5.6:** Effect of uplink bandwidth on the waiting time



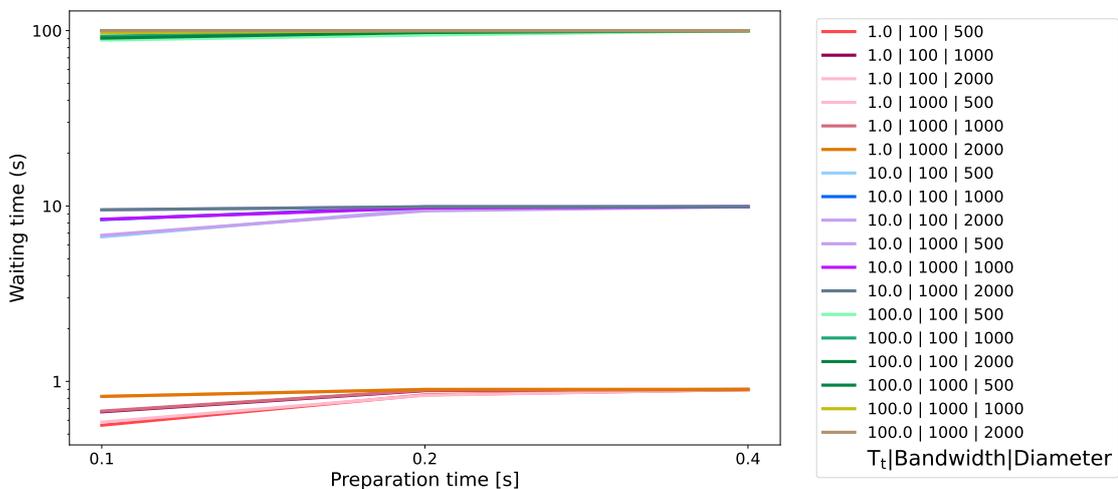
**Figure 5.7:** Effect of uplink bandwidth on the information value

The information value plot on Figure 5.7 has a similar duality to Figure 5.5. Only in this case, the trigger period proves to be the distinctive factor. The higher the

trigger time is, the flatter each function is, culminating in nearly stagnating lines when it reaches its maximum. In terms of diameter an opposite relationship is apparent since as the diameter grows for each trigger time scenario, the lines are developing into steeper functions.

To further dimension our input and constraint parameters, we performed some additional evaluations to determine the effect of the change in  $\mathbb{E}[T_p]$  by adding  $T_t \cdot [0.2, 0.4]$  to its set of values. Note that this parameter is not directly controllable by the system operators or network participants, so we focused only on relevant settings, hence some values and scenarios were omitted.

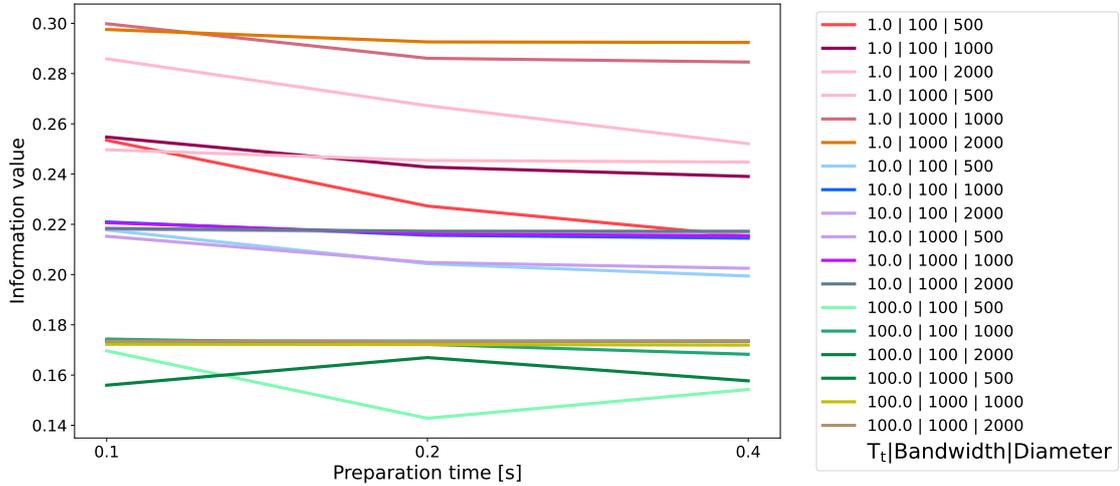
**Preparation time:** In accordance with our previous plots, we specifically examine the effect that the change in the expected value of the preparation time causes on the waiting time and information value. Figures 5.8 and 5.9 both display some similarities to the diameter-related trends, but especially the waiting time metrics. The  $\mathbb{E}[T_p]$  is an immense factor in  $T_w$ , stemming from the definition of the latter. Hence, if  $T_p$  grows then the probability of getting a value much closer to  $T_t$  also increases because the rate of the exponential distribution lowers. Conclusively, as Figure 5.8 unmistakably illustrates, the result rises in  $T_w$ .



**Figure 5.8:** Effect of preparation time on the waiting time

The information value plot is strongly linked to the available uplink bandwidth in each scenario. Figure 5.9 depicts that even medium bandwidth (100 Mb/s) causes a deteriorating trend in information value, while the gigabit bandwidth plots show increasing or stagnating trends, as the  $\mathbb{E}[T_p]$  enlarges. The explanation revolves around the ability of each car to finish as early as possible, given a specific  $T_p$ . If the remaining bandwidth does not further constrain the upload, then the information value is indifferent toward this. On the other hand, if the remaining bandwidth

has become critically low for slow clients with large  $T_p$ , it severely constrains most vehicles. In this case, only a few can upload within  $T_t$ , so the information value drops.



**Figure 5.9:** Effect of preparation time on the information value

In summary, we gathered vast evidence for determining the physical properties of our theoretical model, using numerical simulations. We tightened input and constraint variables to a defined set of values and calculated the results based on these. The findings showed that in a high vehicle density area, such as metropolitan centers, the best setting for both end-to-end latency and information value would be to have HD maps tiles and corresponding 5G cells with a relatively small transmission area ( $\sim 1000$  m in diameter) and as low trigger time as it is sensible for the specific use-case. Additionally, in order to reach as low preparation time as possible, having high bandwidth capabilities in such areas is crucial. These requirements become more tangible than ever, thanks to the growing 5G coverage around the globe, along with the enormous developments that are being carried out in the field of intelligent transportation.

# Chapter 6

## Conclusion

In this paper we presented our proposal for a fast and economical information aggregation and sharing system for vehicular use-cases. Specifically, we tackle the crowd-sourced construction of HD maps that contain up-to-date information about the dynamics of traffic, road conditions, etc. We propose to apply federated analytics for its obvious benefits, and in order to decrease the end-to-end latency and the involved communication footprint, we investigate the performance capabilities of a 5G edge network infrastructure. We provide both theoretical and experimental analysis in the paper for optimizing the scheduling of HD map updates and the number of clients to group together for creating the HD map of a certain area, and we arrive at the conclusion that our proposed system is feasible: the time during which relevant information is delivered to end users at an acceptable scale with 5G capabilities suits the needs of the use-case. To reiterate, the optimal waiting time and information value is achieved by choosing 5G small cells with a diameter of around 1000 m, providing high, Gb/s uplink speed and choosing low but sensible trigger times.

# List of Figures

2.1	Representation of HD map consisting of different layers with static and dynamic information. . . . .	4
4.1	The change of $T_w$ based on network properties . . . . .	12
4.2	The change of $I$ based on $T_w$ and $N_c$ . . . . .	13
4.3	Different parameters after optimization . . . . .	13
5.1	System flow of federated analysis of HD map data . . . . .	16
5.2	Effect of the diameter on the waiting time . . . . .	19
5.3	Effect of the diameter on the information value . . . . .	19
5.4	Effect of the trigger time on the waiting time . . . . .	20
5.5	Effect of the trigger time on the information value . . . . .	20
5.6	Effect of uplink bandwidth on the waiting time . . . . .	21
5.7	Effect of uplink bandwidth on the information value . . . . .	21
5.8	Effect of preparation time on the waiting time . . . . .	22
5.9	Effect of preparation time on the information value . . . . .	23

# List of Tables

4.1	Mathematical notations used in our system model. . . . .	11
5.1	Attributes and their investigated values in simulation scenarios . . . .	18

# Bibliography

- [1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.
- [2] Amazon Web Services, Inc. 5G Edge Computing Infrastructure - AWS Wavelength. <https://aws.amazon.com/wavelength/>, 2022. Accessed: 2022-07-12.
- [3] Daniel J. Beutel, Taner Topal, Akhil Mathur, Xinchu Qiu, Titouan Parcollet, and Nicholas D. Lane. Flower: A friendly federated learning research framework. *CoRR*, abs/2007.14390, 2020.
- [4] Continental AG. Continental Continues to Drive Forward the Development of Server-based Vehicle Architectures. <https://www.continental.com/en/press/press-releases/20210728-cross-domain-hpc/>, 2021. Accessed: 2021-11-30.
- [5] Robert Corless, Gaston Gonnet, D. Hare, David Jeffrey, and D. Knuth. On the lambert w function. *Advances in Computational Mathematics*, 5:329–359, 01 1996.
- [6] George P. Cosmetatos and Gregory P. Prastacos. An approximate analysis of the D/M/1 queue with deterministic customer impatience. *RAIRO - Operations Research - Recherche Opérationnelle*, 19(2):133–142, 1985.

- [7] Ericsson. <https://www.ericsson.com/en/reports-and-papers/ericsson-technology-review/articles/5g-nr-evolution>, 2020. Accessed: 2022-10-21.
- [8] German Aerospace Center. SUMO Documentation. <https://sumo.dlr.de/docs/index.html>, 2022. Accessed: 2022-10-17.
- [9] German Aerospace Center. TraCI. <https://sumo.dlr.de/docs/TraCI.html>, 2022. Accessed: 2022-10-17.
- [10] S. Gomez and J.P. Hennart. *Advances in Optimization and Numerical Analysis*. Mathematics and Its Applications. Springer Netherlands, 2013.
- [11] Google. <https://ai.googleblog.com/2020/05/federated-analytics-collaborative-data.html>, 2020. Accessed: 2022-10-21.
- [12] R. V. Ramos H. B. Banaag, M. S. Litana. Vehicle density estimation in quezon city using object-based feature extraction on satellite images. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLVI-4/W6-202, 2021.
- [13] Birger Jansson. Choosing a good appointment system-a study of queues of the type (d, m, 1). *Operations Research*, 14(2):292–312, 1966.
- [14] Yousef Khalidi. Microsoft partners with the industry to unlock new 5G scenarios with Azure Edge Zones. <https://aws.amazon.com/wavelength/>, 3 2020. Accessed: 2022-07-12.
- [15] Vojdan Kjorveziroski and Sonja Filiposka. Kubernetes distributions for the edge: Serverless performance evaluation. *The Journal of Supercomputing*, 78(11):13728–13755, 2022.
- [16] D. Kraft. *A Software Package for Sequential Quadratic Programming*. Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt Köln: Forschungsbericht. Wiss. Berichtswesen d. DFVLR, 1988.
- [17] Rong Liu, Jinling Wang, and Bingqi Zhang. High definition map for automated driving: Overview and analysis. *Journal of Navigation*, 73(2):324–341, 2020.
- [18] Yang Liu, Tao Fan, Tianjian Chen, Qian Xu, and Qiang Yang. Fate: An industrial grade platform for collaborative learning with data protection. *J. Mach. Learn. Res.*, 22(1), jan 2021.

- [19] A. Loder, L. Ambühl, and M. Menendez. Understanding traffic capacity of urban networks. *Nature Scientific Reports*, 16283, 2019.
- [20] Marquez-Barja, J. and Lannoo, Bart and Naudts, Dries and Braem, B. and Donato, C. and Maglogiannis, Vasilis and Mercelis, S. and Berkvens, R. and Hellinckx, P. and Weyn, M. and Moerman, Ingrid and Latré, Steven. Smart Highway: ITS-G5 and C2VX based testbed for vehicular communications in real environments enhanced by edge/cloud technologies. In *2019 European Conference on Networks and Communications (EuCNC), Abstracts*, page 2. IEEE, 2019.
- [21] Haixia Peng, Qiang Ye, and Xuemin Sherman Shen. Sdn-based resource management for autonomous vehicular networks: A multi-access edge computing approach. *IEEE Wireless Communications*, 26(4):156–162, 2019.
- [22] Amol Phadke. Bringing partner applications to the edge with Google Cloud. <https://cloud.google.com/blog/topics/anthos/anthos-for-telecom-puts-google-cloud-partners-apps-at-the-edge>, 12 2020. Accessed: 2022-07-12.
- [23] Jason Posner, Lewis Tseng, Moayad Aloqaily, and Yaser Jararweh. Federated learning in vehicular networks: Opportunities and solutions. *IEEE Network*, PP:1–8, 02 2021.
- [24] Michael Ulbrich and Stefan Ulbrich. Non-monotone trust region methods for nonlinear equality constrained optimization without a penalty function. *Mathematical Programming, Series B*, 95:103–135, 01 2003.
- [25] Xinze Wu, Xiangming Wen, Luhan Wang, Wei Zheng, Zhaoming Lu, and Lun-ling Liu. A cooperated approach between v2i and v2v for high definition map dissemination in automated driving. In *2020 XXXIIIrd General Assembly and Scientific Symposium of the International Union of Radio Science*, pages 1–4. IEEE, 2020.
- [26] Jinliang Xie, Jie Tang, and Shaoshan Liu. An energy-efficient high definition map data distribution mechanism for autonomous driving. *arXiv preprint arXiv:2010.05233*, 2020.
- [27] Xianzhe Xu, Shuai Gao, and Meixia Tao. Distributed online caching for high-definition maps in autonomous driving systems. *IEEE Wireless Communications Letters*, 10(7):1390–1394, 2021.

- [28] Rongbo Zhang and Kaiyu Cai. The application of edge computing in high-definition maps distribution. In *Proceedings of the 2020 The 2nd World Symposium on Software Engineering*, pages 116–121, 2020.