Attraction-Area based Geo-Clustering for LTE Vehicular CrowdSensing Data Offloading

Douglas F. S. Nunes
University of Sao Paulo, Brazil
douglas@usp.br

Edson S. Moreira
University of Sao Paulo, Brazil
edson@icmc.usp.br

Bruno Y. L. Kimura
Federal University of Sao Paulo, Brazil
bruno.kimura@unifesp.br

Nishanth Sastry
King's College London, UK
nishanth.sastry@kcl.ac.uk

Toktam Mahmoodi
King's College London, UK
toktam.mahmoodi@kcl.ac.uk

ABSTRACT

Vehicular CrowdSensing (VCS) is one of the most emerging and promising solutions designed to remotely collect data from smart vehicles. It enables a dynamic and large-scale phenomena monitoring just by exploring the variety of technologies which have been embedded in modern cars. However, VCS applications might generate a huge amount of data traffic between vehicles and the remote monitoring center, which tends to overload the LTE networks. In big cities, this issue can be even more evident, given the number of vehicles that may roam around. In this paper, we describe and analyze a gEo-clUstering approaCh for Lte vehIcular crowDsEnsing dAta offloadiNg (EUCLIDEAN). It takes advantage of opportunistic vehicle-to-vehicle (V2V) communications to support the VCS data upload process, preserving, as much as possible, the cellular network resources (i.e., channels and bandwidth). In general, it is shown from the presented results that our proposal is a feasible and an effective scheme to reduce up to 92.98 % of the global demand for LTE transmissions while performing vehicle-based sensing tasks in urban areas. The most encouraging results were perceived mainly under high-density conditions (i.e., above 125 vehicles/km^2), where our solution provides the best benefits in terms of cellular network data offloading.

KEYWORDS

LTE Data Offloading, Vehicular CrowdSensing, Vehicle Ad hoc Networks, Vehicular Geo-Clustering, VANET

1 INTRODUCTION

The variety of technologies which have been incorporated in modern vehicles is bestowing them the ability to act as amazing Mobile Sensor Platforms (MSP) [18]. Apart from having an increasing number of sensors, these devices are also being equipped with a greater computing power and different wireless communication interfaces. With such features, this generation of connected vehicles is representing a wide dynamic sensing opportunity to build innovative Intelligent Transportation Systems (ITSs) and a fruitful terrain for developing Vehicular CrowdSensing (VCS) solutions towards the Smart City (SC) [12, 17] era. VCS [14, 18] is one of the most emerging and promising schemes designed to remotely collect data (e.g., fuel consumption, GPS position, engine status and speed) from smart vehicles. It allows for the deployment of powerful monitoring systems just by exploring the technologies embedded in MSPs, with no need for installing extensive infrastructure [5].

Using their native wireless communication capabilities, vehicles can make their on-board data available directly to other local vehicles through Vehicle-to-Vehicle (V2V) ad hoc transmissions, or to the Internet via Vehicle-to-Infrastructure (V2I) transmissions. Such an infrastructure might be twofold: (*i*) computational stations placed alongside the roads, namely Road Side Units (RSUs), which are IEEE 802.11p [10] access points envisioned to assist vehicular communication processes; and (*ii*) LTE radio base stations from cellular networks deployed around. It is important to highlight here that in the vehicular environment, the short-range wireless communications are provisioned by a set of standards stacked into a specific architecture known as IEEE WAVE [20]. WAVE was created to deal with the harsh communication conditions inherent in this scenario (e.g., short contact time, high node speed, and high node mobility), where the traditional Wi-Fi provides a poor experience. When employing local ad hoc transmissions to exchange data, the smart vehicles get into a specific network formation called Vehicular Ad hoc Network (VANET) [19, 21]. Such kind of network is considered a key component to support the building of ITSs and VCS's systems.

Under high-density condition in big cities, the VCS applications might generate a huge amount of traffic between vehicles and monitoring center, tending to overload the network. Once the data upload is usually accomplished via LTE, massive amounts of transmissions can considerably degrade the Quality of Services (QoS) it offers [4]. To deal with this issue, we drew a gEo-clUstering approaCh for Lte vehIcular crowDsEnsing dAta offloadiNg (EUCLIDEAN). EUCLIDEAN is focused on exploring opportunistic V2V communications as a strategy to relieve LTE uplink during VCS data acquisition. Firstly, stationary geo-clusters are created inside the target area. Next, a subset of vehicles is selected to be in charge to gather local information and to report them towards the monitoring center. According to the simulation results of our proposal, in highly crowded conditions up to 92.98 % of vehicles were able

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
MSWiM'17, Nov 2017, Miami, USA
© 2017 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

to report their VCS data samples without the need to access the LTE infrastructure. The two major contributions of this paper are as follows: (*a*) to evaluate the potential reduction of transmissions over cellular network when V2V is additionally used to support the data upload process; and (*b*) to provide a vehicle geo-clustering formation strategy in order to decrease the overall demand for LTE resources in VCS.

The remainder of this paper is organized as follows: Section 2 reviews the main related works; Section 3 describes our proposal; Section 4 goes into finer details with respect to performed simulations; the experimental results are discussed in Section 5; and, finally, Section 6 provides our conclusions and future works.

2 RELATED WORKS

VCS is a compelling mean to deploy monitoring applications in highly dynamic scenarios. This explains why this paradigm has gained evidence in recent scientific research involving vehicular sensing. In line with our proposal, we segmented some of the ITS and VCS related works in two main groups.

The first one is represented by investigations focused on clustering vehicles in a centralized or decentralized fashion [2, 3, 15]. According to [8, 13] cluster-based networking can be considered an attractive solution to provide spatial reuse of the bandwidth, to reduce network congestion, and to simplify message routing or data fusion methods. The cluster formation process reported by the works belonging to this group usually involves several steps and transmissions of additional packets to (*i*) ensure that the created cluster is the best formation; to (*ii*) ensure that the current cluster formation is consistent and updated, dealing with the vehicles' mobility; and to (*iii*) guarantee the election and eventual changes of cluster heads. These cluster formation and maintenance messages may incur in an extra and considerable overhead to the VANET, since the vehicles have a high dynamic mobility pattern and the VCS natively requires multiple ad hoc transmissions to exchange their sensing data itself. To avoid those additional packets, we designed a new geographic clustering approach where the vehicles are able to create clusters and to select their respective cluster heads in a straightforward manner.

The second group brings together studies on provisioning LTE offloading techniques to drain VCS data over alternative networks [4, 7, 15, 16]. The performance results shown by these works demonstrate that the transmissions over cellular radio access system can be drastically reduced when VANET and LTE technologies are combined to support VCS tasks. Our proposal is aligned with the above-mentioned researching, insofar as it integrates some features of those approaches while it extends and creates others for addressing LTE network overload issues. In the majority of the works centered on cellular network offloading, RSUs are fixed on the environment in order to directly collect sensing information from vehicles or/and to forward them to the Internet via a high-speed link. It is important to highlight, however, that this vehicular communication infrastructure is expensive. Deploying them in large scale urban areas, e.g., can be infeasible due to its high cost. We are not considering RSUs in our opportunistic LTE offloading scheme. To relieve the cellular network in crowded conditions, with a high volume of VCS data implied, we explore V2V communications to gather *in situ* sensing information and to reduce the number of LTE transmissions needed for uploading those data to the monitoring center.

3 EUCLIDEAN

The following definitions are used throughout this Section to support the explanation of our proposal:

Definition 1 (Region of Interest)**.** *Region of Interest* (ROI) is a well-defined geographical area (e.g., a portion of a city) whose correlated information is relevant to the sensing applications.

Definition 2 (Attraction Area)**.** In this paper, *Attraction Area* is a logical, circular, and stationary geographic region within an ROI whose centroid is the point where it is expected to find an LTE relay.

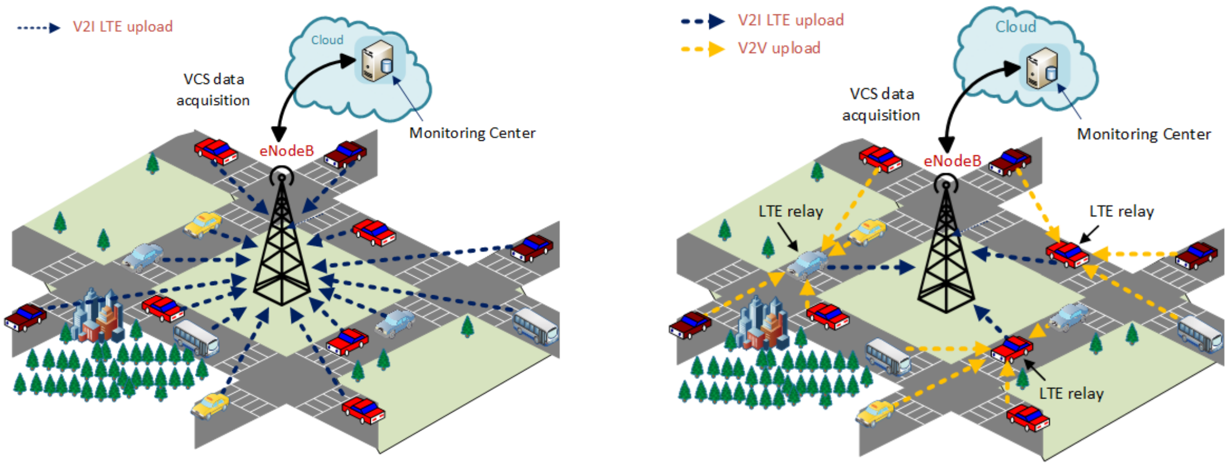
Definition 3 (Geo-Cluster)**.** A *Geo-Cluster* is a grouping of vehicles created within an *Attraction Area* so that they can cooperate amongst one another and exchange their data.

For an illustrative setting, let's consider an urban-like area and a VCS system which is focused on acquiring data about the road traffic itself, via en route vehicles monitoring. Traffic management authorities, drivers, autonomous vehicles or even passengers are the potential stakeholders. In this scenario, we assume that all vehicles are equipped with an On-Board Unit (OBU), a Global Positioning System (GPS) receiver, an LTE cellular network interface, an IEEE 802.11p short-range wireless communication interface, and some built-in machine information sensors. We also assume that the VCS system encompasses a client application version, named as `VCS_TClientApp`, which is deployed on each OBU, and a corresponding server application version, named as `VCS_TServerApp`, which is hosted on a cloud server environment, here called *monitoring center*.

Before starting a data collection, the acquiring task properties need to be configured by the `VCS_TServerApp`. This process consists of assigning values to parameters, such as *data sample collection frequency*, *data report frequency*, *sensing area*, *kind of data required*, and *target vehicles*. According to the VCS application domain, these parameters can assume distinct values. In road traffic monitoring, for instance, a higher *data sample collection frequency* and a higher *data report frequency* are expected, in order to be able to catch the recent road traffic changes. Sensing weather conditions in a large area, on the other hand, may be less strict and may require a lower frequency value for these parameters.

After the acquisition task has its properties defined, an instance of it will be sent out to each desired vehicle roaming in the ROI. Once the task has been received by a target vehicle, the `VCS_TClientApp` will collect the required on-board data and then, under certain predefined conditions (e.g., a time window or a buffer threshold), it will report (upload) them to the `VCS_TServerApp`. This upload operation is depicted in the Fig. 1a, where the data are acquired using only V2I LTE transmissions. This is the main approach adopted by the majority of applications which employ the VCS [16].

When a large number of vehicles is considered to perform VCS tasks (e.g., in big cities), a huge amount of data is expected to require LTE uplink resources in order to traverse the network towards the



(a) data upload using only V2I LTE

(b) data upload using V2V and V2I LTE

Figure 1: VCS data uploading: (a) using only V2I LTE; and (b) using V2V and V2I LTE

monitoring center. Since those LTE resources are finite, this condition tends to significantly degrade the QoS that this infrastructure offers. Under exceptional circumstances, the LTE network might even break down. This is not an unrealistic assumption if we take into account that we are just entering the Internet of Things (IoT) [22] age and that a plenty of devices, besides vehicles, will also compete for LTE resources in the near future. It means that the paradigm illustrated by Fig. 1a does not scale well in very crowded environments. Therefore, in an attempt to reduce this negative impact, we are proposing a strategy to save LTE uplink resources by means of the use of opportunistic V2V communications in addition to the traditional approach presented by Fig. 1a. A pictorial representation of our scheme can be seen in Fig. 1b. Instead of each vehicle uploading its own data directly to the VCS_TServerApp, they will send them to a local subset of vehicles which will be in charge of doing that. The proposed approach, named as EUCLIDEAN, operates over two novel algorithms particularly designed for this purpose: (*i*) Stationary Attraction Areas Allocation's Algorithm; and (*ii*) Attraction Area-based Clustering Protocol.

3.1 Stationary Attraction Areas Allocation Algorithm (S4A)

Finding a subset of vehicles that will be considered LTE relays in VCS systems is not a trivial procedure. Once the sensing is usually performed inside an ROI, defining which vehicles will be selected as LTE relays involves considering where they are located as well. Meeting these two requirements is a challenge, mainly because the VANETs topologies change all the time. Moreover, spreading relays uniformly within the ROI is relevant to guarantee that all vehicles will be able to take advantage of the use of V2V communications to upload their VCS data, thus increasing the potential of saving LTE uplink resources. In this sense, instead of designing a strategy centered on discovering the best vehicle to act as a relay, our scheme is focused on determining where it is expected to find a representative one to drain such data over LTE network.

For this end, we are proposing an effective method based on the concept of logical *attraction area* (See Definition 2). All vehicles inside an attraction area will send their data to a corresponding LTE relay via V2V transmissions, and then it will report that information to the VCS_TServerApp using V2I LTE transmissions, as represented by "Attraction area 1" in Fig. 2.

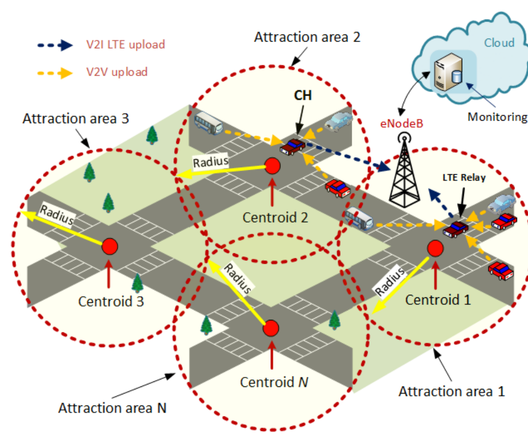


Figure 2: EUCLIDEAN illustrative representation.

Whereas a single LTE relay is expected for each attraction area, we defined a *radius* for it with the same value of the V2V wireless transmission technology range used by vehicles (i.e., according to the IEEE 802.11p interface properties). Thus, if an LTE relay is located just above a given centroid, it will be able to communicate with all other vehicles inside that related attraction area. This assumption is relevant because the attraction area amplitude in EUCLIDEAN implies directly in the number of LTE relays selected. The smaller the attraction area, the greater the number of LTE relays which are expected inside the ROI. However, by reducing the number of LTE relays, we are also reducing the number of LTE transmissions so the benefits of this approach tend to increase. The

LTE relay selection algorithm will be better detailed in the next subsection.

To geo-locate the attraction areas, the S4A first maps all the coordinates of the boundaries of the ROI. Next, it starts to place their centroids side-by-side along the x and y-axis, until the whole ROI is covered. The attraction areas may be totally disjoint among them or they can be distributed in such a way that a given portion of them are overlapped, e.g., they are slightly overlapped in Fig. 2. Therefore, we must provide the allocation algorithm with the bordering coordinates of the ROI and the desired distance amongst the centroids. This can be done by using an interface of the `VCS_TServerApp` with a digital map of the city, by means of clicking and dragging over the area of interest, or via text fields. The S4A was designed to automate the process of determining how much attraction areas will be needed and where they should be placed. It is carried out on the server side, i.e., by the `VCS_TServerApp`, before spreading out the VCS tasks. Performing the S4A on the server side allows the `VCS_TServerApp` to rearrange all the attraction areas at any time. Thus, the vehicle-based sensing becomes even more versatile. After this step, a list of the attraction area centroid coordinates as well as the assumed *radius* will be sent to the `VCS_TClientApps` along with the VCS tasks themselves. Each vehicle will keep these data stored in its local database, for they are imperative to support the geo-clustering formation. A pseudo code description of the S4A is shown in Algorithm 1.

Algorithm 1: S4A

Input: A set $ROI_{bc} = \{bc_{1_{(x,y)}}, bc_{2_{(x,y)}}, \ldots, bc_{n_{(x,y)}}\}$ of coordinates of the boundaries of the ROI; and a positive *distance* value between the centroids.

Output: A set $AA_{cc} = \{cc_{1_{(x,y)}}, cc_{2_{(x,y)}}, \ldots, cc_{n_{(x,y)}}\}$ of *attraction areas* centroids coordinates.

$Coord_y \leftarrow TopMaxCoord_y(ROI_{bc})$;
$i \leftarrow 0$;
while $Coord_y \geq BottomMinCoord_y(ROI_{bc})$ **do**
  $Coord_x \leftarrow LeftMinCoord_x(ROI_{bc})$;
  while $Coord_x \leq RightMaxCoord_x(ROI_{bc})$ **do**
    $cc_{i_{(x,y)}} \leftarrow (Coord_x, Coord_y)$;
    $appendToAA_{cc}(cc_{i_{(x,y)}})$;
    $i \leftarrow i + 1$;
    $Coord_x \leftarrow Coord_x + distance$;
  $Coord_y \leftarrow Coord_y + distance$;
return AA_{cc};

3.2 Attraction Area-based Clustering Protocol (AACP)

As introduced in Section 2, cluster-based networking is an appealing solution to provide spatial reuse of the bandwidth, to reduce network congestion, and to simplify the routing of packets or the aggregation/fusion of data. Considering that these advantages are relevant to the VCS domain, we considered building ephemeral geo-clusters (See Definition 2) of vehicles in our approach. With this grouping formation, the VCS applications can decrease the number of LTE data transmissions needed to perform their sensing tasks. In the AACP, each geo-cluster has associated with it a set of vehicles named Cluster Members (CMs) and a representative one called Cluster Head (CH). The CH is a vehicle timely selected to be responsible for reporting VCS data to the monitoring center on behalf of its one-hop neighbors. In this sense, only CHs are expected to use LTE resources and not all vehicles inside the ROI. Hereafter, the term CH will be used to refer to LTE relay mentioned in the last subsection.

In order to create a geo-cluster, the vehicles make use of: (i) their GPS coordinates; (ii) the GPS coordinates of their neighbors; (iii) the coordinates of the centroids; and (iv) the *radius* of the attraction areas. The own GPS coordinates are obtained from the on-board GPS receiver, the neighbors' GPS coordinates are known via *beaconing* services[1] used by vehicles, and the last two parameters are provided by the `VCS_TServerApp`, as previously mentioned.

Each vehicle $v_i, \forall v \in \mathcal{V}$, keeps a local database with the list of centroids mapped to that ROI, denoted by C, and an updated table with the ID and the current GPS coordinates of all its neighbors, denoted by $\mathcal{N}$. With those data, a vehicle is able to find its nearest centroid, set the geo-cluster it belongs to, and select its corresponding CH in a distributed and straightforward way. The nearest centroid is discovered by means of the Euclidean distance between the vehicle and centroid,

$$d(v_i, c_j) = \sqrt{(v_{ix} - c_{jx})^2 + (v_{iy} - c_{jy})^2}, \qquad (1)$$

where, $\{v_{ix}, v_{iy}\}$ and $\{c_{jx}, c_{jy}\}$ are the GPS coordinates mapped into the Cartesian coordinates of the vehicle v and the centroid c, respectively. The nearest centroid to the vehicle v_i will be that $c_j, \forall c \in C$, with the shortest distance $d_{\min} = \min\{d_1(v_i, c_j), \ldots, d_n(v_i, c_j)\}$. As the AACP assumes that all vehicles roaming up to *radius* meters far from a centroid will belong to the same geo-cluster, by knowing the nearest c_j as well as the current position of its neighbors $\mathcal{N}$, a vehicle v_i becomes conscious of its geo-cluster formation. However, a CH still needs to be defined. A vehicle v_i will be considered a CH of its geo-cluster if it has the shortest distance $d_{\min}$, among its neighbors $\mathcal{N}$, to the corresponding centroid c_j (See *Attraction area* 2 in Fig. 2).

When knowing the coordinates and IDs of its neighbors, a vehicle can calculate by itself the distances of each CM to the nearest centroid. Thus, all vehicles belonging to a geo-cluster are able to identify their respective CH in a distributed way, at any time, without the need to exchange additional messages for this. If CHs move away from their centroids, due to their mobility, new CHs might be selected, once all vehicles inside a geo-cluster are able to become CHs. In this context, EUCLIDEAN considers that centroids are, actually, *attraction points* towards which surrounding VCS data should flow, in the attempt to achieve access to the LTE infrastructure. A pseudo code description of the AACP is shown in Algorithm 2.

During the data collection process, every vehicle periodically performs the following steps:

(s_1) Read the required on-board data samples and insert them into its local `VCS_message_buffer`.

[1]We assume that all OBUs periodically broadcast their identity (ID) and GPS location in special packets denoted *beacons*, with the aim of making the vehicles aware of all its neighbors in quasi real time.

(s_2) Check if it is the current CH. If so, it will send the data samples stored in the `VCS_message_buffer` directly to the remote `VCS_TServerApp`. Otherwise, it will send all data samples to its CH. After receiving an *ack* message from the CH, the vehicle will discard from its buffer those samples that were uploaded via V2V. This latter ad hoc data exchange process employs only single-hop transmissions. To avoid a continuous flip-flop data samples exchange between CHs inside a geo-cluster, when a new one is selected, a maximum buffer length was defined. Every time this threshold is reached, the CH will relief its buffer sending all data samples to the remote monitoring center via V2I LTE.

(s_3) Start a timer to the next reading. Once the timer expires (timeout), go back to the step (s_1).

Algorithm 2: AACP

Input: The vehicle GPS coordinates $v_{i_{(x,y)}}$; a set $\mathcal{N} = \{n_{1_{(x,y)}}, n_{2_{(x,y)}}, \ldots, n_{n_{(x,y)}}\}$ of GPS coordinates from its neighbors; a set $C = \{c_{1_{(x,y)}}, c_{2_{(x,y)}}, \ldots, c_{n_{(x,y)}}\}$ of centroids coordinates; and a positive *radius* value.

Output: A geo-cluster affiliation ID (GCA_{ID}); and the cluster head ID (CH_{ID}).

$nearestCentroid \leftarrow c_{1_{(x,y)}}$;
forall $c_j \in C$ **do**
  $d \leftarrow eucliDistance(v_i, c_j)$;
  if $d < eucliDistance(v_i, nearestCentroid)$ **then**
    $nearestCentroid \leftarrow c_{j_{(x,y)}}$;
if $eucliDistance(v_i, nearestCentroid) < radius$ **then**
  $GCA_{ID} \leftarrow nearestCentroid_{ID}$;
$CH_{ID} \leftarrow v_i$;
$d_{min} \leftarrow eucliDistance(v_i, nearestCentroid)$;
forall $n_i \in \mathcal{N}$ **do**
  $d \leftarrow eucliDistance(n_i, nearestCentroid)$;
  if $d < d_{min}$ **then**
    $d_{min} \leftarrow d$;
    $CH_{ID} \leftarrow n_i$;
return GCA_{ID}, CH_{ID};

Vehicles that were nominated as CH can also aggregate and convert the gathered data into an average value before reporting it to the `VCS_TServerApp`. In this sense, the volume of data uploaded over LTE can be drastically reduced. On the other hand, even that `VCS_TServerApp` asks for a genuine data sample of each vehicle involved with VCS, the advantage of using a single header stack for a set of gathered data is not negligible, as will be shown in Section 5. We adopted this latter model in our experiments. The discussion about the data fusion technique is out of the scope of this work.

4 SIMULATION DESIGN

4.1 Simulation Setup

In order to evaluate the performance of EUCLIDEAN, we created some experiments using a suite of state-of-art simulation tools. The first one was the OMNeT++ [1], an extensible, modular, and component-based C++ framework to support network simulations. The SUMO [11] was employed to model our urban-like scenario and to model our vehicular transport system, i.e., defining vehicles, roads, traffic lights, and the vehicular traffic itself. Lastly, we made use of VEINS LTE [9] to couple the network models provided by OMNET++ with the vehicular traffic models provided by SUMO. VEINS LTE is an open source framework that was created based on OMNET++ and SUMO to carry out vehicular network simulations. A summary of the main network and scenario simulation inputs used by our experiments are shown in Table 1 and Table 2, respectively.

Table 1: Network Simulation Parameters and Values

Parameter	Value
V2V Packet format	WAVE short message
V2V beaconing Frequency	1 Hz
V2V transmission range	500 m
VANET MAC, PHY technology	IEEE 802.11p
802.11p Carrier frequency	5.9 GHz
802.11p Transmission Power	20 mW
802.11p Sensitivity	-89 dBm
802.11p MAC bit rate	18 Mbps
802.11p Radio Propagation Model	Simple path loss model
VCS data report Interval	30 s
Simulation duration time	300 s
LTE transport layer Protocol	UDP
LTE MAC Queue size	2 MB
LTE Carrier frequency	2.1 GHz

So that we could reproduce a realistic urban-like scenario, we decided to use a $\approx 4\,\text{km}^2$ portion of the Bologna city [6] (See Fig. 3). In our experiments, the densities of the vehicles change over time assuming values from 1 to 500; also, the maximum speed reached by them was 70 km/h. The cellular network services are provided by one LTE micro-cell antenna (eNobeB), placed in a central position of the scenario, in such a way it can cover the entire area.

Table 2: Scenario Simulation Parameters and Values

Parameter	Value
Scenario area	$\approx 4\,\text{km}^2$ (1990 m $\times$ 2150 m)
Vehicles densities	$[1, 500]$ vehicles
Maximum vehicles' speed	70 km/h

4.2 Performance Metrics

To assist our analysis as well as to demonstrate the potential of our proposal to reduce VCS data transmission over LTE network, we defined the following performance metrics:

- **Total LTE Packets Transmitted (TLPT)**: it is measured by the sum of all VCS data packets transmitted over the LTE network throughout the data acquisition process;
- **Samples per LTE Transmission (SLT)**: this metric is calculated by using the ratio between the number of VCS data samples received by the `VCS_TServerApp` and the number of LTE transmissions used to report those data;
- **Average Upload Delay (AUD)**: it is measured by the remote `VCS_TServerApp`, computing the average time elapsed between the instant of time when the VCS data were sampled inside the vehicles and the instant of time when they reached the `VCS_TServerApp`; and
- **Number of Cluster Heads Selected (NCHS)**: this metric represents the total of CHs that were selected during all the simulation time.

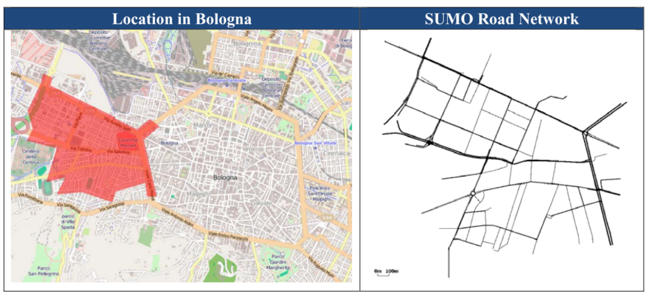


Figure 3: Simulation scenario: area in the city of Bologna [6]

The simulations were performed using multiple runs, with different seed values. We plotted average results whose confidence intervals were calculated with the significance level of 0.05, i.e., assuming 95 % of confidence level.

4.3 Experimental Methodology

For our experiments, it was considered the illustrative scenario previously presented by Section 3. We implemented an instance of the `VCS_TServerApp` which spreads VCS tasks to acquire and report the current GPS coordinates ($v_{i_{(x,y)}}$) and the current speed of the vehicles ($v_{i_{(speed)}}$) every 30 s (this means that both *data sample collection frequency* and *data report frequency* were set to 30 s). The whole Bologna's scenario was assumed to be the ROI and all vehicles were considered by the VCS. To assist the geo-cluster formation, all vehicles periodically (by 1 Hz [2]) broadcast a `HELLO` packet, via beaconing service, containing its ID and its current geographical coordinates. Different vehicle densities (1, 10, 50, 100, 300, 500) [3] and three distinct attraction area centroids' distances (500, 700, 1000) were taken into account by simulations: 500 m for heavily overlapped attraction areas; 700 m for slightly overlapped ones; and 1000 m for totally disjoint ones. The labels "V2V and V2I-500", "V2V and V2I-700", and "V2V and V2I-1000" are used in the next figures for referring to the distances of 500 m, 700 m, and 1000 m, respectively, among centroids. The performance of our LTE offloading strategy was evaluated from individuals' perspectives, mostly comparing it with the traditional pure V2I LTE approach.

5 PERFORMANCE ANALYSIS OF SIMULATION RESULTS

5.1 Traffic reduction from the LTE offloading strategy

Firstly, we were interested in knowing the potential reduction in the number of transmissions over cellular network when V2V communications were also considered in the VCS data upload. The results presented in Fig. 4 show a substantial decreasing in higher density conditions and with slightly overlapped geo-clusters. In this setting, our strategy was able to spend only 7.02 % of the total LTE transmissions used by the pure V2I LTE scheme to report all data samples. Considering an IPv4 header size of 20 bytes (i.e., the shortest) and a UDP header size of 8 bytes, these savings in LTE transmissions while using our geo-cluster approach represent up to 1.03 Mbits[4] of extra data traffic which were not sent over cellular networks. These benefits resulted from the fact that a single header stack is used for a set of gathered VCS data. The aforementioned amount of extra data traffic is not negligible, mostly because it accounts for almost two times the VCS data themselves obtained during the simulation duration time.

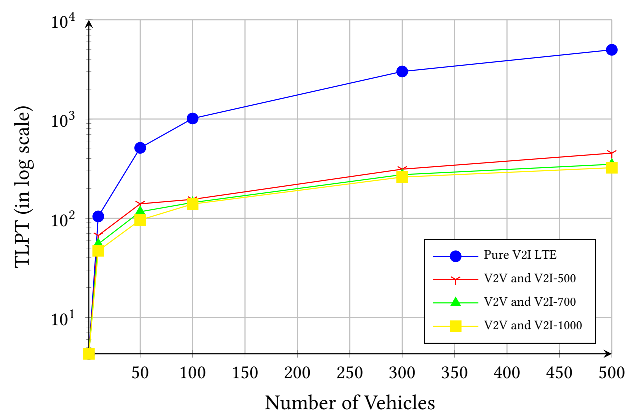


Figure 4: TLPT vs. Number of Vehicles.

5.2 The Impact of Cluster Allocation

We were also interested in understanding how the variations in the clusters allocation process would impact in the upload stage. According to the results depicted in Fig. 5, as expected, the ratio of samples sent to the `VCS_TServerApp` per V2I LTE transmission tends to follow the increase of the vehicles' densities. However, the ratio is relatively reduced when the allocation of heavily overlapping geo-clusters is adopted. Since more CHs are present in this setting, as shown in the Fig. 7, the volume of samples carried by

[2]Other frequency could be used. Evaluating the best one is out of the scope of this work.

[3]This variation aims at representing those situations where the transit is fluent, with low vehicle traffic per km^2, as well as those situations where traffic congestion might be happening. In other words, we aim at evaluating EUCLIDEAN in sparse and high-density vehicular flow conditions.

[4]This means 4635 samples not sent directly via LTE × (20 bytes from IPV4 + 8 bytes from UDP).

each one is smaller. Besides that, the presence of more CHs in the scenario tends to increase the data upload frequency, reducing not only the SLT, but the AUD as well (Fig. 5 and Fig. 6).

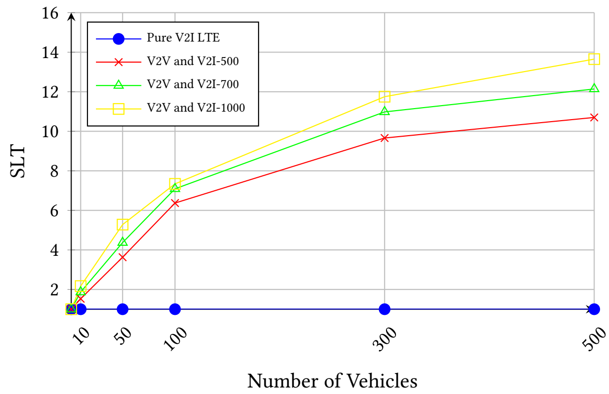


Figure 5: SLT vs. Number of Vehicles.

From the results of Fig. 5 and Fig. 6, we can observe a close relation between the LTE packet payload length, translated into the SLT metric, and their average upload delays. The higher the SLT, the higher the AUD tends to be. The AUD in the geo-cluster approach is also affected by the time the samples remain stored in the `VCS_message_buffers`. When a sample generated by CMs arrives in the CH just after it reported its buffered messages, this sample will be held by the CH until the next report event is triggered. It is important to point out here that, according to the application requirement, the delays can be reduced by means of mechanisms which periodically check the oldest samples in the buffer. If it exceeds a boundary, the CH may anticipate the data upload process.

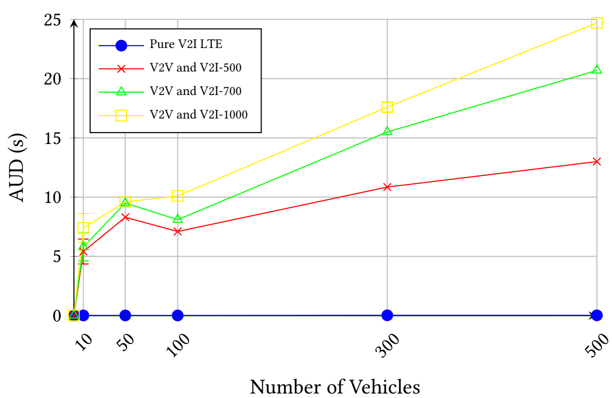


Figure 6: AUD vs. Number of Vehicles.

5.3 The Bunch of Samples

Our next analysis is focused on evaluating how the density of vehicles and the geo-clusters allocation method (disjoint or overlapped) would affect the manner the samples were transmitted in LTE packets. Despite the assumption that the CHs are responsible for gathering local samples and upload them inside a unique LTE packet, we noticed that in many LTE packets received by the `VCS_TServerApp` there was a single data sample. Thus, we make use of the term *bunch of samples* for referring to those LTE packets sent to the `VCS_TServerApp`, whose payloads contained more than a single data sample.

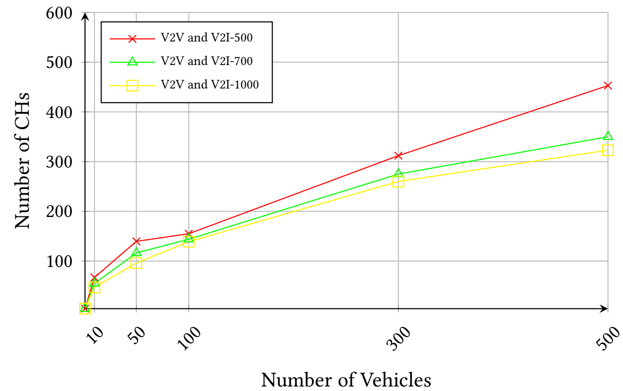


Figure 7: NCHS vs. Number of Vehicles.

The results plotted in Fig. 8 show an ascendant tendency to find bunches of samples in scenarios with a higher density of vehicles. However, they were more present in heavily overlapped geo-clusters. Similarly to the previous observation, this is related to the number of CHs which exists in this condition (see Fig. 7). Comparing these results with those depicted in Fig. 5, Fig. 6, and Fig. 7 we can realize that to obtain smaller end-to-end delays in VCS, it is relevant to consider using EUCLIDEAN with heavily overlapped geo-clusters formation. On the other hand, if the intention is to reduce, as much as possible, the number of LTE transmission, we need to increase the number of samples inside a LTE packet. This requirement can be satisfied by using EUCLIDEAN with slightly overlapped geo-cluster formation or totally disjoint geo-cluster formation, since their results were somewhat similar in higher densities conditions.

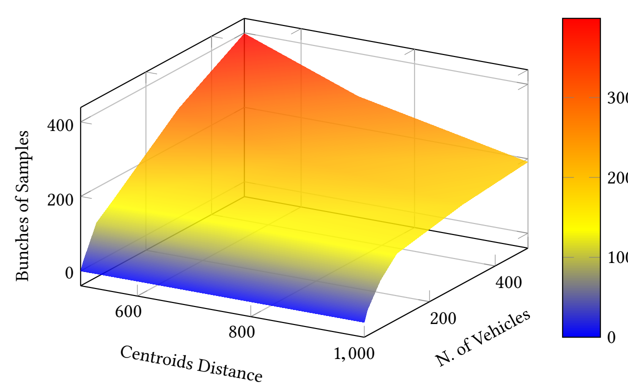


Figure 8: Bunch of samples transmitted over LTE *uplink* vs. Number of Vehicles.

5.4 VCS Samples Size and Storage Occupation

In the presented experiment, each vehicle was scheduled to generate 2 data samples/min. Considering that a data sample is composed by the latitude (4 bytes) and longitude (4 bytes) GPS coordinates values, plus the car speed (4 bytes) value, each vehicle was able to produce 24 bytes/min (accounting for 12 bytes per data sample).

During our simulations, we noticed some CHs carrying up to 130 data samples in their `VCS_message_buffers`. This was the maximum buffer length value reached during our observation; also, it represents a peak of ≈ 2 KB [5] of memory occupation (a vehicle ID (4 bytes) is associated with each data sample stored in `VCS_message_buffer` for retrieving the source of VCS data afterwards). Therefore, it is relevant to point out here that the distributed nature of EUCLIDEAN preserves the storage capacity of the CHs by means of (*i*) scattering the VCS data samples among different CHs and (*ii*) performing its periodic V2I LTE uploads.

6 CONCLUSION

This paper describes and analyzes the EUCLIDEAN: a geo-clustering approach for LTE VCS data offloading. In this scheme, clusters of vehicles are created within specific and stationary regions (*attraction areas*) of the ROI. One CH is then selected for each geo-cluster. The VCS data are locally gathered by CHs and only this kind of vehicle will be in charge to report them towards the *monitoring center*. Thus, exclusively a subset of vehicles can make use of V2I LTE transmissions. As a consequence, the traffic overhead gets shifted from the cellular network to the ad hoc inter-vehicles level.

According to our simulation results, up to 92.98 % of vehicles were able to report their samples without transmitting them directly to the monitoring center. The most encouraging results were perceived mainly under high-density conditions (i.e., above 125 vehicles/km^2. It means that exactly in those situations where the LTE network is prone to be overwhelmed with data transmissions, our proposal provides the best benefits in terms of cellular network data offloading. Future steps will consider deepening the research in cellular offloading techniques. This includes: (*i*) improving EUCLIDEAN so that it can be more dynamic, adapting the amplitude of the attraction areas according to traffic flow changes; (*ii*) incorporating data fusion methods to our approach; and (*iii*) exploiting offloading in LTE downlink as well. Assuming that we are just starting the IoT and SC era, innovative solutions to offload LTE networks, dealing with the massive amount of data generated and required by MSP, will be imperative in the next years.

REFERENCES

[1] 2017. OMNeT++, a Discrete Event Simulator. (apr 2017). https://omnetpp.org/
[2] Hamid Reza Arkian, Reza Ebrahimi Atani, Atefe Pourkhalili, and Saman Kamali. 2014. Cluster-based traffic information generalization in Vehicular Ad-hoc Networks. *Vehicular Communications* 1, 4 (2014), 197 – 207.
[3] M. Azizian, S. Cherkaoui, and A. S. Hafid. 2016. A distributed D-hop cluster formation for VANET. In *2016 IEEE Wireless Communications and Networking Conference*. 1–6. https://doi.org/10.1109/WCNC.2016.7564925
[4] A. Bazzi, B. M. Masini, and G. Pasolini. 2012. V2V and V2R for cellular resources saving in vehicular applications. In *2012 IEEE Wireless Communications and Networking Conference (WCNC)*. 3199–3203.
[5] Alessandro Bazzi and Alberto Zanella. 2016. Position Based Routing in Crowd Sensing Vehicular Networks. *Ad Hoc Netw.* 36, P2 (Jan. 2016), 409–424.

[5] (12 bytes of each sample + 4 bytes of vehicle ID) × 130 buffer entries

[6] Laura Bieker, Daniel Krajzewicz, AntonioPio Morra, Carlo Michelacci, and Fabio Cartolano. 2015. *Traffic Simulation for All: A Real World Traffic Scenario from the City of Bologna*. Springer International Publishing, Cham, 47–60.
[7] G. el Mouna Zhioua, J. Zhang, H. Labiod, N. Tabbane, and S. Tabbane. 2014. VOPP: A VANET offloading potential prediction model. In *2014 IEEE Wireless Communications and Networking Conference (WCNC)*. 2408–2413. https://doi.org/10.1109/WCNC.2014.6952726
[8] Andrea Gorrieri, Marco MartalÃš, Stefano Busanelli, and Gianluigi Ferrari. 2016. Clustering and sensing with decentralized detection in vehicular ad hoc networks. *Ad Hoc Networks* 36, Part 2 (2016), 450 – 464.
[9] Florian Hagenauer, Falko Dressler, and Christoph Sommer. 2014. A Simulator for Heterogeneous Vehicular Networks. In *6th IEEE Vehicular Networking Conference (VNC 2014), Poster Session*. IEEE, Paderborn, Germany, 185–186. https://doi.org/10.1109/VNC.2014.7013339
[10] D. Jiang and L. Delgrossi. 2008. IEEE 802.11p: Towards an International Standard for Wireless Access in Vehicular Environments. In *VTC Spring 2008 - IEEE Vehicular Technology Conference*. 2036–2040.
[11] Daniel Krajzewicz, Jakob Erdmann, Michael Behrisch, and Laura Bieker. 2012. Recent Development and Applications of SUMO - Simulation of Urban MObility. *International Journal On Advances in Systems and Measurements* 5, 3&4 (December 2012), 128–138.
[12] C. Kyriazopoulou. 2015. Smart city technologies and architectures: A literature review. In *2015 International Conference on Smart Cities and Green ICT Systems (SMARTGREENS)*. 1–12.
[13] C. R. Lin and M. Gerla. 1997. Adaptive clustering for mobile wireless networks. *IEEE Journal on Selected Areas in Communications* 15, 7 (Sep 1997), 1265–1275.
[14] Barbara Mavì Masini. 2016. Vehicular Networking for Mobile Crowd Sensing. *Ad Hoc Netw.* 36, P2 (Jan. 2016), 407–408.
[15] P. Salvo, I. Turcanu, F. Cuomo, A. Baiocchi, and I. Rubin. 2016. LTE floating car data application off-loading via VANET driven clustering formation. In *2016 12th Annual Conference on Wireless On-demand Network Systems and Services (WONS)*. 1–8.
[16] R. Stanica, M. Fiore, and F. Malandrino. 2013. Offloading Floating Car Data. In *2013 IEEE 14th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*. 1–9. https://doi.org/10.1109/WoWMoM.2013.6583391
[17] K. Su, J. Li, and H. Fu. 2011. Smart city and the applications. In *2011 International Conference on Electronics, Communications and Control (ICECC)*. 1028–1031.
[18] Julian Timpner and Lars Wolf. 2016. Query-response Geocast for Vehicular Crowd Sensing. *Ad Hoc Netw.* 36, P2 (Jan. 2016), 435–449.
[19] Y. Toor, P. Muhlethaler, A. Laouiti, and A. D. La Fortelle. 2008. Vehicle Ad Hoc networks: applications and related technical issues. *IEEE Communications Surveys Tutorials* 10, 3 (Third 2008), 74–88. https://doi.org/10.1109/COMST.2008.4625806
[20] R. A. Uzcategui, A. J. De Sucre, and G. Acosta-Marum. 2009. Wave: A tutorial. *IEEE Communications Magazine* 47, 5 (May 2009), 126–133.
[21] Yu Wang and Fan Li. 2009. *Vehicular Ad Hoc Networks*. Springer London, London, 503–525.
[22] Feng Xia, Laurence T Yang, Lizhe Wang, and Alexey Vinel. 2012. Internet of things. *International Journal of Communication Systems* 25, 9 (2012), 1101.