LEBANESE AMERICAN UNIVERSITY

On-Demand Mobile Sensing Framework for Traffic Monitoring

By

Sawsan Abdul Rahman

A thesis

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Computer Science

School of Arts and Sciences

May 2017

THESIS APPROVAL FORM

Student Name: Sawsan Abdul Rahman I.D. #: 201401206

Thesis Title : On-Demand Mobile Sensing Framework for Traffic Monitoring

Program: Master of Science in Computer Science

Department: Computer Science & Mathematics

School: School of Arts and Sciences

The undersigned certify that they have examined the final electronic copy of this thesis and approved it in Partial Fulfillment of the requirements for the degree of:

Master in the major of Computer Science

Thesis Advisor's Name <u>A. Mourad</u>	Signature	DATE: <u>11</u> / <u>5</u> / <u>2017</u> <small>Day Month Year</small>
Committee Member's Name <u>Ramzi Haraty</u>	Signature	DATE: <u>11</u> / <u>3</u> / <u>2017</u> <small>Day Month Year</small>
Committee Member's Name <u>Rony Touma</u>	Signature	DATE: <u>11</u> / <u>5</u> / <u>2017</u> <small>Day Month Year</small>

THESIS COPYRIGHT RELEASE FORM

LEBANESE AMERICAN UNIVERSITY NON-EXCLUSIVE DISTRIBUTION LICENSE

By signing and submitting this license, you (the author(s) or copyright owner) grants the Lebanese American University (LAU) the non-exclusive right to reproduce, translate (as defined below), and/or distribute your submission (including the abstract) worldwide in print and electronic formats and in any medium, including but not limited to audio or video. You agree that LAU may, without changing the content, translate the submission to any medium or format for the purpose of preservation. You also agree that LAU may keep more than one copy of this submission for purposes of security, backup and preservation. You represent that the submission is your original work, and that you have the right to grant the rights contained in this license. You also represent that your submission does not, to the best of your knowledge, infringe upon anyone's copyright. If the submission contains material for which you do not hold copyright, you represent that you have obtained the unrestricted permission of the copyright owner to grant LAU the rights required by this license, and that such third-party owned material is clearly identified and acknowledged within the text or content of the submission. IF THE SUBMISSION IS BASED UPON WORK THAT HAS BEEN SPONSORED OR SUPPORTED BY AN AGENCY OR ORGANIZATION OTHER THAN LAU, YOU REPRESENT THAT YOU HAVE FULFILLED ANY RIGHT OF REVIEW OR OTHER OBLIGATIONS REQUIRED BY SUCH CONTRACT OR AGREEMENT. LAU will clearly identify your name(s) as the author(s) or owner(s) of the submission, and will not make any alteration, other than as allowed by this license, to your submission.

Name: Abdul Rahman

Signature:

Date: 11-May-2017

PLAGIARISM POLICY COMPLIANCE STATEMENT

I certify that:

1. I have read and understood LAU's Plagiarism Policy.
2. I understand that failure to comply with this Policy can lead to academic and disciplinary actions against me.
3. This work is substantially my own, and to the extent that any part of this work is not my own I have indicated that by acknowledging its sources.

Name: Saoussen Abdul Rahman

Signature:

Date: 11-May-2017

ACKNOWLEDGMENT

First and foremost, I thank and praise the GOD ALMIGHTY for his Mercies and Grace showered upon me for the successful completion of this work in spite of many challenges faced.

I take this opportunity to express my deepest gratitude to my advisor Dr. Azzam Mourad for his continuous support, motivation, and enthusiasm. This thesis could not have been completed without his generous and professional assistance.

My sincere thanks goes to my thesis committe Dr. Ramzi Haraty and Dr. Rony Touma. They have provided, with kindness, their insight and suggestions, which are precious to me. I am also deeply grateful to the guidance and support of Dr. Mai ElBarachi.

I would like to pay high regards to my beloved parents and for their warm support, which brings belief and hope into my life.

I also reserve special thanks to all my friends whose love and patience have backed me up to keep moving on.

Last but not least, my thanks to all the people who have supported me to complete this research work directly or indirectly.

On-Demand Mobile Sensing Framework for Traffic Monitoring

Sawsan Abdul Rahman

ABSTRACT

With the increased need for mobility and the overcrowding of cities, the area of Intelligent Transportation aims at improving the efficiency, safety, and productivity of transportation systems by relying on communication and sensing technologies. One of the main challenges faced in Intelligent Transportation Systems (ITS) pertains to the real time collection of traffic and road related data, in a cost effective, efficient, and scalable manner. The current approaches still suffer from problems related to the mobile devices energy consumption and overhead in terms of communications and processing. To tackle the aforementioned challenges, we propose in this thesis a novel infrastructure-less on-demand vehicular sensing framework that provides accurate road condition monitoring, while reducing the number of participating vehicles, energy consumption, and communication overhead. Our approach is adopting the concept of Mobile Sensing as a Service (MSaaS), in which mobile owners participate in the data collection activities and decide to offer the sensing capabilities of their phones as services to other users. Unlike existing approaches that rely on opportunistic continuous sensing from all available cars, this ability to offer sensory data to consumers on demand can bring significant benefits to ITS and can constitute an efficient and flexible solution to the problem of real-time traffic/road data collection. Moreover, we extend our approach by elaborating (1) cellular networks based model for selecting suitable set of mobile devices acting as data collectors and (2) inference rules based on deductive logic for traffic status classification inferred from both density and mean speed. A combination of prototyping and traffic simulation traces are used to realize the system, and a variety of test cases are used to evaluate its performance. When compared to the traditional continuous sensing, our proposed on-demand sensing approach provides comparable high traffic estimation accuracy while significantly reducing the resource consumption. This is achieved by selecting the smallest number of data collectors that can provided the best quality of sensed data, in order to maintain a good traffic estimation accuracy and an improved system performance (i.e., lower response time and network load). Other benefits of the proposed on-demand sensing approach include: an overall improved resource efficiency; a better quality of sensed information; more flexible and individual sensing as a service operations; and more users' control over their devices related information.

Keywords: Sensing as a service, Intelligent Transportation Systems, Traffic estimation, On-demand sensing, Road condition, Cellular Tower

Table of Contents

List of Figures	x
List of Tables	xii
1 Introduction	1
1.1 Motivations and Problem Statement	1
1.2 Objectives	5
1.3 Approach Overview and Methodology	6
1.3.1 Sensing as a Service for Traffic Monitoring	7
1.3.2 Traffic Condition Estimation based on On-Demand Sensing with the support of Mobile Infrastructure	10
1.4 Thesis Organization	11
2 Background & Related Work	13
2.1 Introduction	13
2.2 Intelligent Transportation Systems	14
2.3 Traffic Estimation Model	15
2.4 Sensing as a Service	17

2.5	RESTful Web Services	18
2.6	Related Work	21
2.7	Conclusion	25
3	Sensing as a Service for Traffic Monitoring	26
3.1	Introduction	26
3.2	Vehicular Sensing Framework Overview	27
3.2.1	Components Description	28
3.2.2	Web Service based Communication Module	30
3.2.3	Illustrative Scenario and Sequence Diagrams	33
3.3	Participants' Selection and Traffic Estimation Models	35
3.3.1	Matching and Participants' Selection Model	36
3.3.2	Traffic Estimation Model	40
3.4	Prototype and Implementation	42
3.4.1	Prototype software architecture	43
3.4.2	Testbed Setup, Datasets, and Test Scenarios	46
3.5	Experimental Results and Discussion	50
3.5.1	Performance Evaluation Strategy and Metrics	50
3.5.2	Algorithms' Performance Evaluation	51
3.6	Conclusion	62
4	Traffic Condition Estimation based on On-Demand Sensing with the support of Mobile Infrastructure	64
4.1	Introduction	64

4.2	Approach Overview	65
4.3	Vehicle selection Model through mobile Cell Towers	67
4.4	Classification-based Model for Traffic Conditions	70
4.4.1	Density based Estimation	72
4.4.2	Mean Speed based Estimation	73
4.4.3	Rule-Based Inferred Traffic Condition	74
4.5	Solution Validation and Experimental Results	82
4.5.1	Prototype software architecture	82
4.5.2	Testing Scenario	83
4.5.3	Experimental Results	84
4.6	Conclusion	89
5	Conclusion	91

List of Figures

1	High Level Vehicular Sensing System Architecture	27
2	User subscription to platform and user status update scenarios	34
3	On demand traffic condition sensing scenario	35
4	Prototype Components	43
5	Testbed Setup	46
6	On Demand Sensing Scenario	47
7	Continuous Sensing Scenario	48
8	Simulated roads' topology	51
9	Traffic estimation results	51
10	Performance for six/four/three/two criteria of matching algorithm	53
11	Load testing results for on-demand sensing platform	54
12	Data frequency based testing results for on-demand sensing approach	56
13	Data frequency based testing results for continuous sensing approach	57
14	Participation percentage based testing results	59
15	Impact of Selection Criteria on Traffic Estimation Accuracy	61
16	High Level of Fully On-Demand Vehicular Sensing System Architecture	66

17	Representation of the road topology and cellular towers locations on the case study area	68
18	Final traffic condition classified to FF	80
19	Final traffic condition classified to MC	81
20	Final traffic condition classified to TJ	82
21	Full Scenario	83
22	Traffic estimation results	85
23	Comparison of old and new approaches in FF traffic condition	86
24	Comparison of old and new approaches in MC traffic condition	86
25	Load testing results for the full approach	88

List of Tables

1	Web Services Communication Interfaces	31
2	Data representation	32
3	Formulas Notations	36
4	Testing Strategies and Metrics	49
5	Formulas Notations	67
6	Relations between traffic condition and density	72
7	Relations between traffic condition and mean speed	74
8	Boundaries notations for the density and mean speeds	76

Chapter 1

Introduction

1.1 Motivations and Problem Statement

With the rapid widespread of smartphones that come embedded with a variety of sensors (e.g., gyroscope, GPS, and accelerometer), users now hold in the palms of their hands powerful devices that can be used as personal sensing platforms enabling the collection of a wealth of contextual information. This integration of sensing technology in mobile devices opens the door for a new sensing approach and era [1]. Mobile devices can act as super sensors that are readily deployed and can be used to dynamically collect intelligence about cities. There are two main mobile phone sensing paradigms: Participatory sensing in which the user actively participates in the data collection and sensing activity; and opportunistic sensing that occurs in a transparent automated manner without any user involvement [1]. Furthermore, different sensing modes can be adopted, namely: Sense-once (data is collected only one time); Event-based sensing (data is collected and a notification is sent only when an event is detected - e.g. send a notification when the temperature reaches a certain

level); Time-based sensing with expiry duration (data is collected and returned following a regular time interval, until a certain expiry time is reached - e.g., send the user's location each two minutes, from 8 AM until 5 PM); and continuous sensing (data is collected following a regular time interval, without expiry duration - e.g., collect the cars' count in streets all the time).

Sensing technologies constitute one of the key enablers of Intelligent Transportation Systems (ITS). In fact, ITS rely on communication and sensory technologies along with data processing and analysis techniques to improve the safety, efficiency, and productivity of transportation systems [2]. Typical ITS applications include traffic management, road safety applications, and route planning applications. The collection of real time traffic and road conditions constitutes an important challenge in such applications. Conventional methods for the collection of such information typically relied on infrastructure sensors such as surveillance cameras and inductive loops, which may not be always available and involve high deployment and maintenance costs [3]. Recently, the idea of using mobile crowdsensing for the collection of traffic and road related information has attracted attention in academic and industrial forums. In this approach, regular users equipped with sensor-enabled phones collaborate to sense data related to phenomena of interest (e.g. traffic conditions and accidents' occurrence) [4]. The reliance on the drivers carrying sensor-embedded phones for the collection of traffic related information brings important benefits. The first benefit pertains to the easy on-demand deployment of a large-scale network of sensors, since millions of mobile phones are carried everyday by vehicle drivers. Moreover,

this approach leads to important time saving and costs reduction with respect to traditionally deployed specialized sensing infrastructures. Examples of mobile crowdsensing systems used in the area of intelligent transportations include MIT's CarTel [5] and Microsoft Research's Nericell [6]. These systems mainly adopt a continuous sensing approach in which data is continuously sampled from all cars on all street segments (without the explicit involvement of users), and then processed offline on the backend server. However, this imposes high energy-requirements on mobile devices, entails significant overhead on the mobile communication infrastructure, and results in large amounts of data requiring processing on the server. Furthermore, the opportunistic automated data collection strategy adopted by such systems gives rise to privacy concerns by mobile users, which may not wish to share sensory data that reveals sensitive information about themselves (e.g. their geographic location).

Moreover, in similar context, the connected vehicles technology [7] has emerged recently, which enables the communication between vehicles (i.e. Vehicle to vehicle) as well as between vehicles and the roads' infrastructure (i.e. vehicle to infrastructure), using dedicated short range communications (DSRC). Despite the merits of the connected vehicles technology and its potential use for safety and congestion management applications, this technology presents certain limitations when compared to the mobile crowdsensing technology. The first limitation pertains to the smaller market penetration rate of connected vehicles (fore casted to reach 152 million connected vehicles sold by 2020 [8]), when compared to the massive and pervasive market penetration of smart-phones that have passed the 2 billion device mark in 2016 [9]. In fact, the effectiveness of ITS relying on sensory technology depends on the sufficient penetration of the technology in streets, a fact

that cannot be currently guaranteed with connected vehicles, but can be easily achieved with smart-phones. Furthermore, smart vehicles currently face limitations in terms of their communication and sensing capabilities, under adverse weather conditions.

Recently, the Mobile Sensing as a Service (MSaaS) approach has been emerged [10], in which mobile devices and users willingly participate in the sensing process and offer their phones' sensory data collection capabilities as services to other users. This approach is very promising to address the aforementioned issues, and to the best of our knowledge, none of the previous related works consider it in ITS solutions. In this work, we propose a novel vehicular sensing framework enabling on-demand road condition monitoring in efficient and flexible manner. Unlike existing solutions that rely on opportunistic continuous sensing from all cars available, we advocate participatory on-demand sensing from a selected number of cars that can offer a high quality of sensed information. Many interesting scenarios could be enabled by the concept of on-demand sensing as a service. In the sequel, we provide two participatory and opportunistic transportation related scenarios:

- **On-Demand Accident Scene Intelligence Gathering:** When an accident occurs, it takes the police some time to arrive to the accident site. In order to collect information about the accident before arrival to the site, the police force could send an on-demand sensing request that would be conveyed by the sensing platform to a selected group of cars in the accident area. The car drivers who accept this request would then take pictures/videos of the accident as well as collect additional contextual information using their phones and push them back to the platform. The platform would then process this data and produce a summary report containing information

such as the number of stationary cars, number of casualties/people laying on the floor, and temperature/smoke levels within the accident scene. This summary report along with collected pictures and videos footage would be returned by the platform to the police force for fast situation assessment and decision-making.

- **On-Demand Road Condition Monitoring:** Drivers on the road could serve as source of information for traffic and road conditions, by using their phones to collect contextual information such as snow removal conditions, potholes in streets, fog or bad weather conditions, accidents, extreme traffic, and road redirection. This information could be requested in real time by drivers heading in a certain direction and wishing to learn about the roads' conditions in order to either continue on a specific road or find an alternative one. In this case, a driver would send an on-demand road condition-sensing request to the sensing platform. This last would forward the request to a set of targeted cars located in the specified destination, get the required data as their responses and then process it and send the response back to the requester. This way, the data consumer would be able to gather useful real-time information about roads' conditions, and thus reach his/her destination within a short trip time. Such scenario applies to both participatory and opportunistic sensing paradigms.

1.2 Objectives

The goal of this thesis is to address the problems related to crowd-sensing for traffic condition monitoring. More precisely, our main objectives can be listed as follows:

- Reduce the energy consumption on devices and communication overhead on mobile infrastructure caused by the continuous crowd-sensing for traffic condition estimation.
- Maintain accurate traffic condition estimation while reducing as much as possible the number of participating mobile devices and offering them the option of willingly contributing to the sensing process.

1.3 Approach Overview and Methodology

In this thesis, we propose an infrastructure-less on-demand vehicular sensing framework to provide accurate estimation for traffic condition. The proposed approach adopts the concept of MSaaS, which brings significant benefits to ITS and constitutes an efficient and flexible solution to the problem of real-time road data collection. Adopting such concept helps reducing the number of participating vehicles, energy consumption, and communication overhead. Moreover, we extend our work by (1) enhancing the selection of the suitable set of mobile devices on the roads through the use of cellular network, and (2) classifying the traffic conditions inferred from both density and mean speed through inference rules based on deductive logic.

In order to study the performance of our vehicular platform and compare the on-demand sensing approach to the traditional continuous sensing approach, we combined prototyping and simulated traffic traces to build a proof-of-concept prototype of the system. Furthermore, we conducted extensive experiments in which different parameters were varied, such as: the traffic conditions on the road, the matching criteria used for participants' selection,

the number of sensing requests received by the platform/hour, the frequency of voluntary data publication requests, and the percentage of cars participating in the sensing activities. Four main performance metrics were measured using various test cases, namely: The traffic estimation accuracy, the participants' selection accuracy, the system's response time, and the system's network load. This comparative performance analysis gives interesting insights on the contributions and benefits of an-demand participatory sensing approach, data collection frequency, percentage of cars participating in the data collection activity, traffic estimation accuracy, and system's performance.

In the sequel, we describe in details the aforementioned contributions.

1.3.1 Sensing as a Service for Traffic Monitoring

In this work, we developed a partial on-demand sensing framework based on Sensing as a Service approach in order to estimate the mean speed of a particular road. Our approach encompasses three components: Data consumers requesting the road traffic in any area of interest, data collectors offering the sensing capabilities of their phones as services, and the vehicular traffic platform acting as data broker between the consumers and collectors to process the sensed data and predict the roads traffic status. In this approach, traffic data sensing about any region of interest would occur on demand, when triggered by a sensing request. Once the sensing request is received by the sensing platform from a data consumer, the set of targeted users acting as data collectors will be determined by the platform by assuming that each collector shares every two minutes its recent sensed data with the platform. To best select the set of data collectors, we propose a multi-criteria

matching algorithm that takes into account the collectors' presence in the region of interest, their phones sensing capabilities, the users' willingness to participate in the sensing activity, the users' reputation, the phones' battery level, and the accuracy of the data they provide. Once the sensed data is received from the targeted data collectors, the sensing platform relies on a mean speed estimation algorithm to estimate the mean speed on the specified road, which is sent to the user who sent the original sensing trigger request.

The main contribution of this work is monitoring the roads while adopting the concept of Mobile Sensing as a Service. The related achievements are summarized as follows:

- **High Mean Speed Estimation Accuracy:** Our proposed approach is able to infer mean speeds close to the ground truth (i.e., real mean speed) in all the tested scenarios. The experimental results show a small variation in the calculated mean speed values compared to the ground truth. Moreover, experiments explore that the estimation error % decreases from 17.82% for 10 sensing requests received/hour to 2.9% for 10000 sensing requests received/hour in the on-demand approach. Such results are comparable to the standard continuous approach, in which the estimation error decreases from 17.7% for data voluntarily published each 10 minutes to 5.9% for data published each 30 seconds.
- **High Quality of Sensed Information:** Our approach relies on a multi-criteria selection approach that enables the achievement of a high Quality of sensed information, by selecting the best candidates yielding the highest quality records satisfying multiple quality of information criteria. This participants' selection approach leads to more accurate traffic estimation results.

- **Reduced Resource Consumption:** Important reduction in resource consumption can be achieved such as the amount of generated network load, energy consumption on mobile devices, and amount of data requiring processing on the server. Unlike the traditional continuous sensing where data is collected in terms of seconds, participating users in our platform share every two minutes their sensed data and thus resource consumption is significantly reduced. Moreover, our proposed approach strikes a balance between traffic estimation accuracy and resource consumption. This is achieved by using contextual information and a multi-criteria participants' selection approach to select the smallest number of data collectors that can provide the best quality of sensed data, in order to maintain a good traffic estimation accuracy and an improved system performance (i.e. lower response time and network load).
- **Users' control over their devices related information:** Our participatory sensing approach offers more control to mobile phone users over the sensed data collected using their devices, since users can accept or deny a sensing request. This is not the case in opportunistic continuous sensing which is typically performed systematically, without the involvement/consultation of users.
- **Flexible and Individual Sensing as a Service Operations:** In the case of continuous sensing, street maps showing traffic conditions may not be available for all countries and all cities, as it requires agreements with telecommunication authorities. On the other hand, the idea of on-demand sensing allows more flexibility and availability of data in any area of interest through individual mutual agreements (i.e. sensing as a service upon the user consent), by tapping into the sensing capabilities of millions of

mobile phones deployed across the globe to obtain traffic conditions, in any street of interest.

1.3.2 Traffic Condition Estimation based on On-Demand Sensing with the support of Mobile Infrastructure

Some limitations arise when considering our first proposed approach. Users, specially data collectors, must share every two minutes their recent sensed data with the server, which limits our full on-demand approach within Intelligent Transportation System context. Moreover, the proposed traffic estimation algorithm estimates the mean speed on a particular road which may not reflect a clear image about the roads traffic condition for the requester. Therefore, we extend our framework to address those limitations by embedding new models for both matching and traffic estimation existing modules in addition to a new one for analysis and reporting. In the matching module, we rely on the cellular towers that cover the area of interest to get the set of collectors whenever a data consumer sends a sensing request in order to overcome the periodic sensing from mobile devices. As for the traffic estimation module, we rely on the density characteristic besides the mean speed to best reflect the traffic condition. Furthermore, the new module analysis and reporting is implemented to classify the traffic into Free Flow, Moderate Congestion, or Traffic Jam from the calculated density and mean speed based on inference rules.

The main achievement of this work is the fully on-demand approach through the support of cellular networks. In addition to the previous contributions, this work adds the following contributions for the proposed framework:

- **High Traffic Estimation Accuracy:** Our proposed approach is able to successfully infer the traffic status (i.e. free flowing, moderately congested, and traffic jam) in all the tested scenarios based on combination of mean speed and density. The experimental results show 100% accuracy for the classification of the traffic condition.
- **More Resource Consumption Reduction:** Due to the fact that the data collectors offer their sensed data on need basis only (instead of continuously or periodically publishing their information) and only a chosen number of phones is selected based on several selection criteria, significant and more reduction in the amount of generated network load, energy consumption on mobile devices, and amount of data requiring processing on the server is achieved.

1.4 Thesis Organization

The remainder of the thesis is organized as following:

In Chapter 2, we give an overview on the concepts of Intelligent Transportation Systems, Traffic Estimation Model, Mobile Sensing as a Service and Web Services. Afterwards, we present some of the related works in the fields of continuous sensing for the collection of traffic in addition to Sensing as a service by mobile phone sensors.

In Chapter 3, we present our on-demand vehicular sensing platform for the real-time road condition monitoring. First, we reveal the framework architecture with a full description of its different components and entities. Then, we describe how, when, and what type of messages are exchanged among the system components. Finally, we conducted some

experiments to evaluate the performance of our proposed approach and compare it with other approaches.

In Chapter 4, we extend the elaborate approach to support fully on-demand strategy. First, we describe a new method based on cellular tower to collect the most suitable set of collectors. Then, we propose a combined classification approach based on density and mean speed to classify the traffic condition into three phases. Finally, we present the experimental results and discuss the performance evaluation of the approach.

In Chapter 5, we summarize the thesis by recapitulating the contributions and drawing the future work directions.

Chapter 2

Background & Related Work

2.1 Introduction

This chapter is devoted to present several concepts that form our model. The goal of the proposed on-demand vehicular platform is to estimate the traffic flow which is the objective of one of the Intelligent Transportation Systems (ITS) applications. Thus, we give a definition of ITS and describe their different technologies and applications. Moreover, the traffic flow is estimated by calculating the mean speed of a specific road and then computing the density in order to classify the traffic condition into Free Flowing, Moderate Congestion or Traffic Jam. Therefore, we provide an overview about different Traffic Estimation models and how each characteristic used to visualize the traffic stream is obtained. The collection of traffic and road related information is achieved through sensed data gathered from mobile phones located in the area of interest. We present hence the Mobile Phone Sensing as a Service approach and how mobile phones and users participate in the sensing activities. Finally, we summarize the current literature on the approaches in the areas of continuous

sensing for the collection of traffic and sensing as a service by mobile phone sensors. We also highlight their limitations and the need to our contributions.

2.2 Intelligent Transportation Systems

Intelligent Transportation System is an emerging yet challenging system which entails the interaction with vehicles, road operators, and drivers through wireless technologies. The vehicle-to-vehicle (V2V) or vehicle-to-infrastructure (V2I) communication involved in ITS applications brings significant improvement in terms of efficiency, safety and productivity of transportation systems.

Various technologies and applications are used in Intelligent transportation systems. The technologies can be grouped as follows: 1) Wireless Communications such as WiMAX, GSM, or 3G for long-range communications and IEEE 802.11 protocols for the short ranges of few hundred meters. However, the use of Mobile ad hoc networks when transmitting information from one vehicle to the next one can extend the short-range communications. 2) Global Positioning System (GPS) that is embedded recently in an increasing number of vehicles allows to determine the position of the vehicle through the signal received from different satellites. 3) Probe Data such as location and speed gathered from the so-called "probe vehicles" usually deployed in taxis. Probe data are reported to a Traffic Management Center to be analyzed and to generate the average travel speed on the different roads. 4) Sensing technologies embedded in the infrastructure systems such as video vehicle detection and inductive loop detection to detect the vehicles that pass through a specific point.

As for the ITS applications, they can be classified in the following categories: 1) Advanced Traveler Information Systems that provide valuable information for drivers such as transit information, traffic regulation and incidents. 2) Advanced Transportation Management Systems that improve the road traffic flow and reduce the congestion through the Transportation Management Center. 3) ITS-Enabled Transportation Pricing Systems that are used in the transportation systems of the funding countries. 4) Advanced Public Transportation Systems that attract the public transportation riders by providing them with the real-time statuses of buses and trains such as their arrival and departure time in the automatic vehicle location (AVL) applications. 5) Vehicle-to-infrastructure and vehicle-to-vehicle integration in transportation systems that involve the communication between vehicles to roadside sensors or vehicles to vehicles.

2.3 Traffic Estimation Model

Traffic flow models are helpful guidance to deploy Intelligent Transportation Systems. One of the technologies of ITS enables probe vehicles to report to Traffic Management Center some sensed data that could be analyzed to generate roads traffic flow and identify the congested area. There are microscopic and macroscopic traffic flow models. In microscopic model, the behavior of single vehicle in the stream is studied. While in macroscopic model, the whole traffic stream is considered. Moreover, density, mean speed and flow [11] are three main characteristics used to visualize a traffic stream and the ITS applications rely on for traffic management.

- Density (k) is defined as the number of vehicles occupying a roadway segment in a

specific time and is represented as follows:

$$k = \frac{N}{L}$$

where N is the number of vehicles, L the of the roadway and k is expressed in units of vehicles/distance .

- Speed representing the distance traveled per unit time has an average computed in two ways in a traffic stream:

1. Time Mean Speed (v_t) is the average speed of all vehicles passing a reference point on the roadway over a duration of time, and is given by

$$v_t = \frac{1}{n} \sum_{i=1}^n v_i \quad (1)$$

where n represents the number of observed vehicles passing the reference point and v_i the spot speed of i^{th} vehicle.

The Time Mean Speed is generally measured through Loop Detectors that can detect vehicles crossing a determined point and can pursue their speeds.

2. Space Mean Speed (v_s) is the average speed of all vehicles passing a given roadway segment. It is based on average travel time of each vehicle traversing a segment of a roadway and is denoted as follows

$$v_s = \frac{n}{\sum_{i=1}^n \frac{1}{v_i}} \quad (2)$$

where n represents the number of observed vehicles and $\frac{1}{v_i}$ the time the vehicle i takes to traverse a roadway segment.

The Space Mean Speed is measured from cameras and/or satellite pictures.

- Flow (q) represents the number of vehicles crossing a reference point over a duration of time and is represented as follows

$$q = \frac{N}{T} \quad (3)$$

where N is the number of vehicles counted and T the elapsed time.

Flow rates are usually expressed in units of vehicles per hour but the actual measurement interval is much less, representing a flow for a period of 15 mins, 1 min, 30 secs, etc. The Flow can be calculated from loop detectors that track vehicles passing a reference point over time.

2.4 Sensing as a Service

Nowadays, the ubiquitous mobile phones not only perform as key entertainment and communication devices, but also come embedded with a rich set of sensors of seven sensors per device on average [12]. These sensors enable interesting sensing applications across a wide variety of domains such as transportation, healthcare, homecare, environmental monitoring, ecommerce, social networks and safety [13]. To encourage users to participate in the sensing activities and use their sensor capabilities in order to build a cloud computing model that uses the collected sensed data and provide diverse sensing services is the key

concept of Sensing as a Service (SaaS). When a cloud user sends a sensing request from a phone, laptop or desktop to a typical SaaS cloud, a sensing server receives the request, forwards it to a set of mobile phones located in the area of interest, stores the data and sends it back to the requester cloud user.

Two mobile phone sensing paradigms exist: Participatory Sensing and Opportunistic Sensing.

- In Participatory sensing, users carrying mobile phone actively engage during the sensing activities and manually involve in the sensing action to determine what, where, how, and when to sense such as taking some picture for an event or a certain location.
- In Opportunistic sensing, the mobile phone involves in making decision during the sensing activities and decides whether to send data and store it or not without any user involvement.

2.5 RESTful Web Services

Web services can be described as software module executing one or many tasks that can be delivered over a network as per the World Wide Web Consortium (W3C) and are not restricted to specific operating system or programming language. One type of web services is RESTful Web service which follows Representational State Transfer (REST) protocol. It is used everywhere and it is even the chosen method in the big Internet companies like Google, Amazon and Facebook.

In REST, everything including functionality and data is considered as a resource. A resource is accessed through Uniform Resource Identifiers (URIs) and uses the standard HTTP methods: GET, PUT, POST, and DELETE to read, update, create, and delete operations respectively. Moreover, REST is an architectural style enabling services to work best on the web through its six described constraints:

- Uniform Interface: Rest services should be designed following the uniform interface, which simplifies the architecture and has the following four constraints:
 - Resource-Based: This states that the requests should identify the individual resources looking for such as the use of URI standard. Additionally, the resources themselves may be different from how they are returned to the client (i.e. receiving data from the server in the form of HTML, JSON or XML)
 - Resource Manipulation through Representation: This states that the client manipulates the representation of resources by requesting a specific representation which fits his need (i.e. requesting a JSON or XML representation of a resource)
 - Self-descriptive messages: This states that the messages sent from the client should include all the required data to describe how to act on the resource.
 - Hypermedia as the engine of application state (HATEOAS): As any interaction with a resource is stateless, meaning that each pair of request/response is independent from any previous pair, a response message includes in the response body hyperlinks to other available actions.

- Client-server: The clients are separated from servers and by separating them, clients will no longer be concerned with data storage as it remains internal on the server side. Therefore, the components could be replaced and evolved independently. Such constraint improves not only the portability of the client code, but also the scalability of the server components.
- Stateless: Client/Server interaction is always stateless in REST web services. Each pair of request and response is independent of the latest one where the server does not store anything related to the previous request that the client made. Instead, the server considers every request as new and therefore the client must include all needed data in the request for the server to fulfill it.
- Cacheable: In order to avoid clients from reusing stale data in the second fetch and hence reducing the total network traffic, resources must declare themselves cacheable. Caching improves both the performance on the client side and the scalability on the server side.
- Layered System: REST supports the use of layered system, known as intermediates, such as proxies and gateways. They are used for load balancing, enforcement of security policies and response caching, and thus helps improving the system scalability.
- Code on demand (optional): Executable code rather than XML/JSON representations can be returned to the client if required.

2.6 Related Work

Several works on sensing for the continuous collection of traffic and road related information have been recently carried out. In this approach, a group of users having sensor-enabled devices (e.g. mobile phones, GPS readers) collectively sense relevant data to estimate the traffic condition in a specific area of interest. Moreover, there is a rich literature providing traffic estimation approaches that rely on specialized sensing devices embedded into Intelligent transport infrastructure. In the following, we elaborate the main related approaches in addition to the technical problem statement at the end of the section.

In [14], the authors proposed the use of GPS and accelerometer data for the detection of traffic conditions, abnormalities, and potholes on roads. This approach consists of five components: smartphones, a local database (for temporary storage of data), open wireless networks, a server hosting a central database, and open street maps. The sensed data is sent to a heuristic algorithm that analyzes it and produces roads' traffic status. Herring et al. [15] proposed a solution that targets traffic conditions on highways. The model consists of one physical component which is the GPS, and three cyber components: a cellular network operator, cellular phone data aggregation and traffic service provision, and traffic estimation algorithms. In this approach, data is collected using mobile phones on specific trajectories called virtual trip lines. This data is sent to a server that aggregates it and sends it to the Ensemble Kalman Filtering based traffic estimation algorithm. In [16], Thiagarajan et al. proposed an approach to overcome energy consumption and inaccurate position sampling challenges by using a Hidden Markov Model (HMM) that depicts the trajectory of a vehicle over a portion area in the map. They performed map matching in order to estimate the

travel times of the traversed road segments. In [6], Mohan et al. proposed a solution called NeriCell that focuses on the sensing component such as accelerometer, microphone, GSM radio, and GPS sensors. They used Intelligent Traffic System that needs dedicated sensors in streets and cars. This solution consists of a system of rich monitoring of road and traffic conditions that piggybacks on smartphones and calculates roads' traffic status using vehicles' acceleration data. Herreraa et al. proposed two data gathering techniques (spatial and temporal) in [17]. Spatial sampling implies that equipped vehicles report their information (position, velocity, etc..) at specific time intervals T regardless of their positions, while temporal sampling implies that the vehicles report their information as they cross some spatially defined sampling points. In this approach, data is collected from mobile devices (Nokia N95) every three seconds, then the instantaneous velocity is measured at the same rate, and these data will form a rich history of data used for traffic estimation. Also, they targeted and solved the privacy aspect concerning the identity of the users. Recently, the authors in [18] proposed a distributed peer-to-peer approach to traffic estimation. In this approach, a car uses V2V communication to collect position and velocity related data from nearby cars. The data collected is sparse data in the form of floating car data snapshots and the Underwood traffic-engineering model based on density is used for traffic condition estimation.

In another context, the following proposed approaches were focusing on the importance of sensing as a service by mobile phone sensors: Ban and Gruteser, in [19], focused on two important issues. The first one is fine-grained urban traffic knowledge extraction, while the second is the privacy protection scheme. They provided a comparison between the primary way of collecting through fixed-location sensors and the newly suggested one

through the mobile phones sensors. Based on their claims, collecting data through fixed-location sensors costs a lot and is not an efficient way in order to predict traffic efficiently, while collecting and extracting data through mobile phones will greatly benefit the urban traffic prediction applications in terms of performance. This type of collecting data can provide detailed behaviors and continuous trajectories of the vehicles. In [20], Khan et al. conducted a survey that talks about the different monitors and usages of mobile sensing which are: health, traffic, environment, social, special purpose, human behavior, and commerce. It mainly distinguishes between two types of urban sensing. The first type is the participatory, while the second is the opportunistic. In both types of sensing, the solutions implemented are divided into three main parts: personal, public, and social. In each of the solutions, authors emphasize the used type of sensors, hardware and software description, communication modules, and applications. In [21], Das et al. did not target the traffic estimation on roads problem, rather they focused on the community sensing (participatory and opportunistic). They focused on the community sensing which targets the embedded sensors on the mobile phones such as GPS, camera, audio, accelerometer, and GSM. The main goals of their paper are to ensure (1) generality by supporting a wide range of applications with flexibility of reusing existing code, (2) security by ensuring that the participating phones belonging to individual users remain secure and that the applications do not misuse sensitive sensor information, and (3) scalability by allowing the system to scale to a large number of nodes without placing an undue burden on the infrastructure itself. In [22], Placzek focused on the idea of reducing the amount of data transmitted through Vehicular Sensor Network in order to control the roads traffic. Instead of periodically requesting sensed data from vehicles, the proposed approach specifies time moments when the queries

should be sent. The selected time moments are characterized by the uncertainty of traffic estimation, and in this case, new traffic data is requested.

Problem Statement: All the aforementioned mobile sensing related approaches rely on continuous or periodic sensing of road and traffic data, which entails the following problems:

- High energy consumption on mobile devices due to the continuous sensing from the relevant sensor such as GPS, accelerometer, etc.
- Communication overhead on mobile infrastructure due to (1) continuous data sensing from each vehicle and (2) data collection from all the vehicles without any filtering/selection criteria during collection. All the customizations performed by these approaches are done at the server side, i.e. during traffic analysis after collection. The impact of this problem will potentially increase with the fast emergence of Internet of things that will overhead the mobile infrastructure, reaching around 50 Billion connected devices by year 2020 [23].
- High processing overhead and full availability of road data since the current traffic analysis models and algorithms are dependent on continuous and complete data collection from all vehicles in order to estimate the mean speed, density, and flow.

In order to address the aforementioned problems, the proposed framework offers on-demand and upon need data collection gathered based on several selection criteria such as availability, location, need, etc. To the best of our knowledge, none of the current approaches in the literature have targeted the aforementioned problems and addressed on-demand sensing in the context of ITS and traffic estimation.

2.7 Conclusion

We presented in this chapter brief descriptions about Intelligent Transportation Systems, Traffic estimation models, Sensing as a Service approach and RESTful Web services. Additionally, we presented an overview of the major approaches in the literature in the areas of Continuous sensing for the collection of sensed data and Sensing as a Service concept in the mobile devices.

Chapter 3

Sensing as a Service for Traffic Monitoring

3.1 Introduction

This chapter addresses the problem of continuous sensing for road and traffic data to estimate the traffic condition by proposing an on-demand sensing in the context of ITS. Using the concept of Sensing as a Service, a data consumer could request a specific road condition from the sensing platform. This last would search for the most suitable set of cars located and/or heading toward the targeted road through a matching algorithm based on multi-criteria. Afterwards, the platform collect the sensed data from the selected set to run a traffic estimation algorithm that calculates the mean speed of the requested road.

The rest of the chapter is organized as follows. Section 3.2 details the proposed vehicular sensing framework, by presenting the components functionalities, describing the web service communication interfaces, and illustrating the operation using some scenarios.

Section 3.3 is dedicated to the description of the proposed traffic estimation and participants selection models. This is followed by the prototype and implementation in section 3.4 and the experimental results in Section 3.5. We end this chapter with our conclusions, in Section 3.6

3.2 Vehicular Sensing Framework Overview

Figure 1 depicts the high-level architecture of the proposed vehicular sensing framework.

Our system encompasses three main roles: Data consumers interested in the acquisition

Figure 1: High Level Vehicular Sensing System Architecture

of sensed data related to a particular area of interest within the city (e.g., provide me with traffic conditions or snow clearance conditions on road X); data collectors offering their phones' data collection/sensing capabilities as services to other users; and vehicular sensing platform acting as intermediary and data broker between consumers and collectors.

The vehicular sensing platform receives sensing requests from data consumers and matches those requests with the most suitable data collectors based on some selection criteria. Afterwards, the platform sends the sensing request to the chosen data collectors through the matching model, who can either accept or reject it. Those who accept the request would perform the required sensing task and send the sensed data to the vehicular sensing platform, which is responsible of validating, aggregating and processing it through the relevant model and algorithms, and then sending the reply to the requestor. The communication between the different roles can occur either using mobile communication infrastructures (e.g. 3G/4G mobile networks) or over public WiFi hotspots if available (e.g. in smart cities).

3.2.1 Components Description

We now describe the functions performed by our system's entities in more detail:

- **Data Consumer:** The data consumer is a user who is interested in sensing services. To access those services, the data consumer interacts with the vehicular sensing platform through a gateway application to discover the sensing communities available. Once subscribed to a sensing community, the data consumer can discover and subscribe to (all or some of) its associated services. An example of a sensing community could be "New York city drivers" and examples of sensing services are "Traffic condition monitoring service" and "Snow clearance notification service". After subscribing to sensing services, a data consumer can send a sensing trigger to the vehicular sensing platform by specifying the requested data type and sensing mode (i.e. sense once, event-based sensing, or continuous sensing), as well as the geographical area

of interest.

- **Data Collector:** A data collector is a user equipped with a sensor-enabled mobile device, and who is willing to offer its data collection capabilities as services to other users. The mobile device should host a sensing gateway application enabling the interaction with the vehicular sensing platform. To offer sensing services, a data collector must first subscribe to become part of a sensing community. After subscription, the data collector periodically publishes his/her availability to the sensing platform (e.g. available, busy, and away) to indicate willingness to participate in sensing activities. The data collector's sensing gateway application should support a number of functionalities, including: handling sensing trigger requests from the platform; allowing the user to initiate sensing without trigger (i.e. offer-based sensing) and send the captured data to the sensing platform; ability to collect requested data from embedded sensors; supporting some information processing and formatting capabilities; providing Geo-temporal tagging of the sensed information; scheduling of sensing tasks; and management of sensing sessions based on received requests.
- **Vehicular Sensing Platform:** The vehicular sensing platform constitutes the key entity in our architecture. It acts as intermediary between data consumers and data collectors by matching sensing requests (in real time) with the most suitable data sources, and offers information management and data brokerage capabilities. To achieve that role, the vehicular sensing platform consists of a number of modules, namely: *communication, storage, validation, matching, identification and traffic estimation*. The *communication module* is responsible of creating the communication

messages (requests and responses) exchanged between the platform and the users. The *storage module* is responsible for storing sensing activities related information. The *validation module* is responsible of the pre-processing of the collected information to detect inconsistencies and calibrate data. The *matching module* is a key module implementing a matching algorithm that relies on certain criteria (e.g. location and availability of data collector, data collection capabilities, data accuracy, available battery level, and user's reputation) to match sensing requests with the most suitable set of data collectors. The *identification module* is responsible of assigning unique IDs to the sensed entities, the sensing services offered, as well as users' roles in the system. Finally, The *traffic estimation module* processes the raw sensed data and produces traffic status information based on the proposed traffic estimation model and algorithm. We provide in the following sections the technical details of the models and algorithms deployed in the sensing platform.

3.2.2 Web Service based Communication Module

The communication module in the sensing platform is responsible of handling the messages among the system entities. In the following, we present in details the type of the exchanged messages.

In our vehicular sensing system, the communication between the components should be flexible and light weight since the platform supports multiple sensing requests and handles their response in parallel. Therefore, we select RESTful [24] as it is the best to work for mobile Web services and web-based applications. Representational State Transfer (REST)

Table 1: Web Services Communication Interfaces

Resources	URI Base URL: http://VehicularSensing.com	HTTP action/description
Sensing Session	/SensingSession /SensingSession /{SensingSessionID}	POST: create a new sensing session GET: get all sessions GET: retrieve a session. PUT: update a session. DELETE: terminate a sensing session
Data Consumer	/DataConsumers /{DataConsumerID}	GET: get info about a data consumer. PUT: update a data consumer info. DELETE: remove a data consumer from a session
Data Collector	/DataCollectors /DataCollectors /{DataCollectorID}	POST: create a new data collector GET: get all data collectors GET: get info about a data collector. PUT: update data collector's status info. DELETE: delete a data collector
Traffic Report	/SensingSession/SensingSessionID/traffic	GET: get traffic report related to a sensing session PUT: update traffic report info. DELETE: remove a traffic and dissociate it from session

is an architectural style where data is considered as resources and accessed through Uniform Resource Identifiers (URIs).

Each entity in our system is thus considered as a web service that communicates through REST APIs. The exchanged messages, illustrated in Table 1, use the HTTP protocol and its most commonly used operations: POST, GET, PUT, and DELETE. POST creates a new resource and its URI will be automatically generated. GET reads the information about the resource in an appropriate representation. PUT updates the resource that can be deleted

through DELETE. In the proposed model, our data set consists of Sensing Sessions split into four resources: "SensingSession", "DataConsumer", "DataCollector" and "TrafficReport". Each SensingSession has one DataConsumer willing to get the traffic status of a particular road and a set of DataCollectors located on the specified road and a TrafficReport generated after processing the sensed data. The "SensingSession" resource is identified by the URI "http://VehicularSensing.com/SensingSessions/SensingSessionID" where SensingSessionID is the unique identifier of the Session. The "DataConsumer" is identified by "http://VehicularSensing.com/DataConsumers/DataConsumerID" where DataConsumerID is the identifier of the requester. The "DataCollector" is identified by "http://VehicularSensing.com/DataCollectors/DataCollectorID". The "TrafficReport" is identified by "http://VehicularSensing.com/SensingSessions/SensingSessionID/traffic". Table 1 summarizes also the URIs used in column 2, along with their operations in column 3 in order to access each resource found in the first column.

Table 2 illustrates the data encompassed in the request/response messages exchanged between the client and server for the HTTP actions related to the SensingSession resource.

Table 2: Data representation

Resources	HTTP action	Data representation Operation	
		Client ->Server	Server ->Client
Sensing Session	POST: http://VehicularSensing.com/SensingSessions	<SensingSensing> <datatype>Traffic </datatype> <dcName>Name1</dcName> <areaofInterest> <city>cityZ </city> <street>Street79 </street> </areaofInterest> </SensingSensing>	http://www.VehicularSensing.com/SensingSession/{Sensing SessionID}

GET: get all sessions http://VehicularSensing.com/SensingSessions	None	<pre> <SensingSessions> <SensingSession> <ssID>ID123</ssID> <dcName>Name1</dcName> <datatype>Traffic</datatype> <areaofInterest> <city>cityZ</city> <street>Street79</street> </areaofInterest> </SensingSession> </SensingSessions> </pre>
GET: retrieve a session http://VehicularSensing.com /SensingSession/{Sensing SessionID}	None	<pre> <SensingSession> <ssID>ID123</ssID> <dcName>Name1</dcName> <datatype>Traffic</datatype> <areaofInterest> <city>cityZ</city> <street>Street79</street> </areaofInterest> </SensingSession> </pre>
PUT: update a session http://VehicularSensing.com /SensingSession/{Sensing SessionID}	<pre> <SensingSession > <datatype>Snow Condition</datatype> </SensingSession> </pre>	None
DELETE: terminate a sensing session http://VehicularSensing.com /SensingSession/{Sensing SessionID}	None	None

3.2.3 Illustrative Scenario and Sequence Diagrams

Figure 2 represents a sequence diagram that shows the interaction among the system components. Whenever a user wants to subscribe to the vehicular sensing platform, he indicates his role as either data consumer or data collector and sends a POST request to the server. Accordingly, the server creates new "DataCollector" / "DataConsumer" resource and sends him back the appropriate URI. Once the resource is successfully created, a "200 Ok" message is returned to the user. As the server requires to periodically keep in its database the

Figure 2: User subscription to platform and user status update scenarios

sensed data of all participating data collectors, each one sends a PUT message to the server specifying its current position, speed and availability.

The diagram in figure 3 shows how a data consumer can request the traffic status of a particular road. First, the requester of data sends a POST message to the server in order to create new sensing session and gets the generated URI resource as a response. Then a GET request is sent from the consumer to the server holding the created session id as shown in step 4 in the figure. When the server receives the request, it runs an internal matching algorithm to collect the expected targeted cars on the road and sends them GET requests. If available, every data collector replies by sharing its current position and speed with the server, which runs another matching algorithm to filter the on-road cars only. Finally, the server platform generates the road's condition and sends it to the user after processing the sensed data and applying a traffic estimation algorithm.

Figure 3: On demand traffic condition sensing scenario

3.3 Participants' Selection and Traffic Estimation Models

The key models in our Vehicular Sensing platform are the matching/participant selection model and the traffic estimation model. We focus in these models on the scenario where a data consumer sends a sensing trigger request to the sensing platform, with the following parameters: Data type = traffic condition; sensing mode = sense once; Area of interest = name of street on which sensing is required. Once the user sends the request, the server first runs the algorithm realizing the Matching model to retrieve the most suitable set of data collectors along with their sensed data. Then it runs the algorithm implementing the Traffic Estimation model to process the raw sensed data and predict the traffic status. All the notations of the used formulas in these two models are illustrated in Table 3.

Table 3: Formulas Notations

Variable	Description
R	The desired road from which the traffic condition is inferred
R'	An adjacent road heading toward R
R''	An adjacent road heading from R
pos_l^s	Last position of sensor s at time t_1
pos_{cur}^s	Current position of s at time t_2
$s.avail$	The availability of s
$s.rep$	The reputation of s
$s.capab$	The capability of s
$s.dataAcc$	The data accuracy of s

3.3.1 Matching and Participants' Selection Model

The matching model is needed to retrieve the appropriate set of collectors whenever the platform's server receives a sensing request. In this context, several models have been advanced to select the suitable set. The participatory sensing framework proposed in [25] selects the social sensors and enables to share data based on their availability, trust and energy. To predict the user location and estimate his availability, an algorithm called Dynamic Tensor Analysis (DTA) is adopted since the user historical trajectory is known through his daily routine. All users with similar trajectories are clustered in 'Friends-Like Social Sensors' group where only one is selected to avoid the same data collection from multiple participants. The same concept has been proposed in [26] on how to choose the best set from a huge number of collectors and retrieve sensing data from them. The model focuses on finding not only the best set but also the minimum number of participants in the set

that covers a given area of interest and satisfies certain constraints. The sensing requests can be sent at any time and handle both temporal and special requirements. The authors in [27] cover a certain area of interest based on the budget constraint by focusing on the scenario where the entire targeted region is divided into several sub-regions. The participants in each sub-region set specific prices in order to respond to the sensing requests and thus the system picks the ones with lowest prices to maximize the number of collectors. Some interpolation methods could be used in case the incentive budget is not sufficient or no collectors are located in the desired sub-region. In [28] and [29], the data consumer sends sensing tasks to the system server where several requirements are associated to the tasks such as the sensing area, time, data granularity and quantity. The proposed selection models in [28] and [29] allow to gather the maximum number of sensory collectors while minimizing the consumption of energy for all the participants. However, all of them did not target ITS and Traffic condition monitoring, which limit their relevance to our approach. Moreover, they are missing many important criteria needed for a traffic decision model.

In this context, we propose in the sequel a new matching model that considers several criteria for selecting the minimal set of sensing vehicles. The first criterion is the geographic location of the targeted collectors, which takes into consideration two cases; in the first one the data is collected from the cars located on the targeted road, while in the second where the data is collected from the cars that are heading toward the desired road and will eventually be located on it after a certain time frame. In order to find out the position of each participating collector without nullifying the on-demand sensing concept, each data collector must share periodically its recent sensed data with the server. Accordingly, the first case can be calculated since both the positions of the participating nodes and the road

coordinates are predefined. However, concerning the second case, we determine a bounding circle around the middle of the targeted road and take the nearby streets that fall in this circle area based on the map topology. All cars located on the nearby streets are then added to the set of data collectors. The second criterion in the matching is the availability of the user. The status of the user is checked whether available or not to recognize if he is willing to participate in the sensing activities. At the time a user sends a non-availability, the server should not consider him in the set of collectors even if he is located in the desired area of interest. The third criterion is the battery level of the users' mobile phone. If the phone battery level of a user is less than or equals to 20%, then our matching approach assumes that the mobile phone is not capable of sending/receiving any form of data to/from the server and thus the user is not in the appropriate set of collectors. The fourth criterion is the user's reputation, which helps to improve the performance of the platform. We keep the records of data sent by the user in order to check its accuracy. For example, if a user history has low accuracy, we won't send him new requests. The fifth criterion is the sensing capabilities of the user. This is useful in the general case where the user is sensing data related to temperature, CO2 level, or any other type of information. We need to check if the user is capable of sensing such type of data since not all the phones embed variety of sensors. The sixth and final criterion is the accuracy of the data sent by the user based on the type of the used sensor. For instance, the data collected using GPS is considered more accurate than the one collected using Wi-Fi.

In the sequel, we present the model combining all the aforementioned criteria followed by its corresponding algorithm (Algorithm 1). All the participating sensors S with total

size n are first selected as input to find the initial S_{init} defined by

$$S_{init} = \sum_{s=1}^n (pos_l^s \in R \vee pos_l^s \in R') \wedge (s.avail == true) \wedge (s.BatLev \geq 20\%) \wedge (s.rep == high) \wedge (s.capab == true) \wedge (s.dataAcc == good) \quad (1)$$

where S_{init} set holds all the sensors which their last position pos_l^s at time t_1 was either on R or heading toward R , and are characterized by the following properties: are available, have high reputation, capable to sense the required data, and have good data accuracy. Let t_1 be the time when the participants shared the last sensed data with the server just before receiving a traffic request from a consumer and t_2 be the request time.

Since the server requires two recent sensed data for the collectors to estimate the roads conditions, the server sends sensing requests to each car in the list and gets their new positions and speeds as response. Hence, once the collectors in the set S_{init} are found, the platform sends them sensed request to collect the appropriate data in order to estimate the road condition. The new set of sensors S_{final} collected after receiving S_{init} ' responses is defined by

$$S_{final} = \sum_{s=1}^{S_{init}.size} (pos_l^s \in R \wedge pos_{cur}^s \in R) \vee (pos_l^s \in R \wedge pos_{cur}^s \in R'') \vee (pos_l^s \in R' \wedge pos_{cur}^s \in R) \quad (2)$$

where set S_{final} contains the cars that are located on R at time t_2 and were located on R' at time t_1 , the cars that are located in the destination R at both t_1 and t_2 , and the cars that were found on the road R at t_1 and becomes on its adjacent R'' at t_2 . Accordingly,

the cars, heading toward the desired destination and changed their directions, are removed from the list and the rest will be sent to the traffic estimation module.

Algorithm 1 - Matching

```

1: Input: All cars  $s_i$  participating in sensing services
2: Output: Set of targeted cars  $S_{final}$  located in the specified destination
3: Construct a list  $S_{init}$ :  $\emptyset$  for the estimated targeted cars
4: for each sensor  $s_i$  do
5:   get  $s_i$  last position  $pos_l^s$  from the server's database
6:   if  $pos_l^s == \text{onRoad} \parallel pos_l^s == \text{headingToRoad}$  then
7:     if  $s_i == \text{available}$  then
8:       if  $\text{batteryLevel} \geq 20\%$  then
9:         if  $\text{reputation} == \text{high}$  then
10:          if  $\text{capability} == \text{true}$  then
11:            if  $\text{dataAcc} == \text{good}$  then
12:              add  $s_i$  to  $S_{init}$ 
13: Construct a new list  $S_{final}$  for the targeted cars
14: for each sensor  $s_i$  in  $S_{init}$  do
15:   send sensing request  $r_i$  to  $s_i$  and get its current position  $pos_{cur}^s$ 
16:   if  $pos_l^s == \text{onRoad} \&\& pos_{cur}^s == \text{onRoad}$  then
17:     add  $s_i$  to  $S_{final}$ 
18:   if  $pos_l^s == \text{onRoad} \&\& pos_{cur}^s == \text{outOfRoad}$  then
19:     add  $s_i$  to  $S_{final}$ 
20:   if  $pos_l^s == \text{headingToRoad} \&\& pos_{cur}^s == \text{onRoad}$  then
21:     add  $s_i$  to  $S_{final}$ 

```

3.3.2 Traffic Estimation Model

Once the vehicular platform successfully performs the matching process, the platform forwards the two sensed data pos_l^s and pos_{cur}^s for each sensor s in the set of collectors S_{final} to the traffic estimation module in order to estimate the speed on the specified road on which

the traffic condition is inferred. Since the sensors have varied positions on the map, some of them may have pos_l^s located on the specified road, while others located on its adjacent roads, as the set of collectors encompasses the cars heading toward the desired destination. Similarly, at time t_2 , when the server sends the sensing request to the set of selected data collectors, the sensors' pos_{cur}^s could either be on the specified road or on its adjacent one in case it left it. The distance of road traveled by sensor s is denoted as $r_i(pos_l^s, pos_{cur}^s)$, which takes only the distance traveled within the two intersections of the road without the adjacent links as the data consumers ask for the condition of a specific road. Knowing $r_i(pos_l^s, pos_{cur}^s)$ of each sensor s_i during the interval (t_1, t_2) , the server can compute their average speed v_i defined by

$$v_i = \frac{r_i(pos_l^s, pos_{cur}^s)}{(t_1, t_2)} \quad (3)$$

The road condition represented by the mean speed is calculated according to equation (4) [30]

$$v_{mean}(t_2) = \frac{\sum_{s \in S_{final}(t_2)} [v_i \times r_i(pos_l^s, pos_{cur}^s)]}{\sum_{s \in S_{final}(t_2)} r_i(pos_l^s, pos_{cur}^s)} \quad (4)$$

where the mean speed v_{mean} of a particular road at the request time t_2 is a function of the length of the road traveled and covered by each sensor s in the final set S_{final} , along with their average mobile speed.

Note that this approach is widely used for traffic speed estimation, and works under the assumption that vehicles' speeds are constant.

Typically, the ground truth (v_{GT}) is calculated using video surveillance of real traffic,

which is a statistical measure that describes the entire traffic flow as follows

$$v_{GT}(t_k) = \frac{l}{\frac{1}{|C_i(t_k)|} \times \sum_{c \in C_i(t_k)} \Delta t_c} \quad (5)$$

We use visual observation of the traffic simulation to determine a set of cars $C_i(t_k)$ that enter the road segment within a certain time window $(t_1, t_2) \subseteq (t_k - \tau, t_k + \tau)$, where t_k is the chosen moment in time to calculate the ground truth and τ is a predefined constant. For those set of cars, we calculated the time taken by each one of them (Δt_c) to traverse the road segment of length l .

To determine the accuracy of the obtained results, we calculate the estimation error using equation (6) that represents the absolute value of the calculated mean speed minus the ground truth

$$\bar{E} = |v_{mean} - v_{GT}| \quad (6)$$

3.4 Prototype and Implementation

In order to validate our proposed solution, we combined prototyping with simulation traces generated using VanetMobiSim, which is a widely used traffic simulator that generates realistic vehicular movement traces, based on macroscopic and microscopic mobility models [31]. Instead of using real sensory data collected using phones, we opted for simulation traces as it allows the generation of a large set of data for our experiments and enables the control of different parameters (e.g. roads' topology, number of cars used, mobility model, and speed limits on the roads). In our experiments, we used a macroscopic mobility model

that deals with properties such as traffic density, speed and flow.

3.4.1 Prototype software architecture

Figure 4: Prototype Components

Figure 4 illustrates our prototype software architecture. The prototype, which was implemented in JAVA, consists of three main components: a data consumer node generating sensing trigger requests; a vehicular sensing platform node matching requests with collectors, managing the sensed data and estimating traffic status; and data collector nodes responding to the sensing requests and publishing their sensed data. Communication between the different components is achieved using REST APIs. To simplify the development of RESTful Web Services, we have selected the open source Jersey framework [24] that functions as a JAX-RS Reference Implementation [32], and Grizzly Application server

that deploys the web services. Each component encompasses a PostgreSQL repository [33] to store the relevant sensed data.

As shown in Figure 4, the data consumer is a node consisting of a request/response handling module responsible of the generation of sensing requests and the handling of responses; a sensing session manager responsible of the tracking of the sensing sessions and their status; and a local sensing data repository (SDR) storing the collected data and the sensing sessions' statuses.

The vehicular sensing platform is the main node in our prototype. It consists of the following modules: a request/response handler responsible of the processing of received requests and responses; a validation and matching module implementing the matching algorithm and validating the data received; a request dispatcher and request queue responsible of queuing and dispatching requests to selected data collectors; a resource naming module responsible of assigning IDs to sensed entities, sensing services, and users; a publication engine handling voluntary data publications from data collectors; a traffic estimation module implementing the proposed traffic estimation algorithm; an analysis and reporting module responsible of the generation of advanced traffic reports from the collected data; and a sensing data repository (SDR) storing the sensed data, the generated traffic reports, the sensing sessions' status as well as information about data collectors and consumers.

Instead of hosting the data collector node logic on real mobile devices, we used instances of the data collector nodes running on one machine to simulate a large number of data collectors. Furthermore, we used VanetMobiSim to simulate different traffic conditions (i.e. the positions and speeds of the cars moving on the simulated roads), and stored this information in a file that was made accessible to the data collectors' instances. This

file, which contains information related to all simulated car nodes, is initially processed by each data collector node to retrieve its specific information throughout the simulation lifetime, and stored on a local DB on the node in question. This combination of prototyping and simulated traffic data allows testing at different scales, not to mention the control of the traffic parameters that would not be possible with a real life prototype deployed on smart phones hosted in moving vehicles.

To achieve the functionality of a data collector, each data collector node consists of a request/response handling module responsible of receiving the sensing requests and publishing their sensed data; a sensing data publication module responsible of publishing the sensed data to the platform (either following a trigger or voluntarily); a sensing session manager module responsible of the tracking of the sensing sessions and their status; a scheduler module responsible of scheduling the processing of multiple requests received from the platform; an info acquisition module responsible of the retrieval of the car position and velocity sensed data (at that specific time instance) from the PostgreSQL local database (the SDR) hosted by the data collector node; an information processor module responsible of processing and formatting the messages exchanged via REST API between the vehicular sensing platform and the data collector node; and a SDR that stores all the data collector position/velocity information throughout the lifetime of the simulation, to be used whenever the platform asks for data collection. It should be noted that the information stored in the SDR is used either to publish data voluntarily to the sensing platform without any solicitation and trigger, or used to respond to sensing requests by sending the vehicle's velocity and position at certain time instance to the platform, thus covering two modes of information publication (trigger based publication and voluntary publication).

Figure 5: Testbed Setup

3.4.2 Testbed Setup, Datasets, and Test Scenarios

As shown in Figure 5, the experimental setup consists of three main components: One data consumer node triggering the sensing requests, one vehicular sensing platform node responsible of data and sensing requests/responses management and implementing the matching and traffic estimation algorithms, and one data collector management node that instantiates the needed data collector instances and dispatches sensing requests to the relevant ones. The used machines are equipped with Intel Core™2 Duo E6550, 2.33GHz processor and 4GB of RAM, 10000 RPM HDD, 100MBPS link and running Ubuntu 12.04 LTS.

To populate the raw data repository accessible to the data collector instances, four data

Figure 6: On Demand Sensing Scenario

sets were generated using VanetMobiSim simulations. The simulation runs were configured to simulated four traffic conditions, namely: free flowing, moderately congested, congested, and highly congested. Furthermore, in order to compare our proposed on-demand sensing approach to the traditional continuous sensing approach, two test scenarios were used in our experiments as illustrated in Figures 6 and 7.

In the on-demand sensing scenario depicted in Figure 6, the first interaction is triggered by the data consumer, which sends a sensing request to the vehicular sensing platform asking for the traffic condition in an area of interest (i.e. specific position or street). The sensing platform will then run the matching algorithm to get the list of suitable data collectors satisfying the matching criteria and forward to them the sensing request. Each data collector will perform the sensing operation (i.e. acquiring its position and speed in that case) and sends the sensed information as a response to the sensing platform. After receiving the responses from all targeted data collectors, the sensing platform will run the traffic

Figure 7: Continuous Sensing Scenario

estimation algorithm to estimate the traffic speed/condition. This information is then used to build a traffic report, which is sent by the sensing platform as final response to the data consumer.

In the continuous sensing approach depicted in Figure 7, the sensing operation is performed in a continuous fashion by all the data collectors, which publish their sensed information on a regular basis to the sensing platform. When a data consumer sends a sensing request to the platform, the latter uses the sensed information previously published to estimate the traffic condition using the traffic estimation algorithm, and then sends the final response (i.e. the traffic report) to the data consumer.

Table 4: Testing Strategies and Metrics

Test Category	Performance Metric	Description of how metric was measured/calculated	Test scenarios used
Traffic Estimation Algorithm	- Mean speed - Ground truth - Traffic estimation Error	Mean speed: calculated using equation 4 Ground truth: calculated using equation 5 Estimation error: calculated using equation 6	Four scenarios were used: 1. Free flowing road 2. Moderately Congested road 3. Congested road 4. Highly Congested road. The simulated traffic data for those scenarios was generated using Vanet-MobiSim by varying the configuration of the max. speed on the road. Figure illustrates the road topology used in the simulated scenarios.
Matching Algorithm	Response time	Time needed for the matching algorithm to return the set of selected cars located in the area of interest and matching the specified matching criteria.	Four variants of the matching algorithm were tested by varying the dataset (i.e. the # of cars processed during the selection) and the matching criteria used. The four variants are: 1. Six matching criteria (Proximity, availability, Data collection capability, accuracy, battery level, reputation) 2. Four matching Criteria (Proximity, availability, Data collection capability, accuracy) 3. Three matching Criteria (Proximity, availability, Data collection capability) 4. Two matching Criteria (Proximity & availability) The dataset for each experiment was crafted in a way to show the difference between the different matching criteria. For instance, to highlight the effect of reputation as matching criteria, we introduced malicious nodes that injected wrong data in the data set & nodes, which would be filtered out only if reputation is used as matching criteria. For accuracy, we introduced data that is rounded and not very accurate. This approach allows the differentiation between the different versions of the multi-criteria matching algorithm, and to show the trade-off between performance and accuracy.
	Matching error %	Matching error %: calculated as # of cars selected by algorithm / # of cars satisfying the matching criteria (calculated manually) * 100	
System Load Testing	Response time	Time from when sensing request (msg. 1 in fig. 6) is sent until sensing response (msg. 7 in fig. 6) is received.	Using the on-demand sensing scenario presented in Fig. 6, we varied the number of requests sent by the data consumer to the platform from 1 to 2000 requests, and measured the system's response time and network load.
	Network Load	Size of packets exchanged for the end-to-end interaction (between sensing request and sensing response)	
System Data Frequency Based Testing- Continuous Sensing Approach	Response time	Time from when traffic condition request (msg. 4 in fig. 7) is sent until traffic condition response (msg. 5 in fig. 7) is received.	Using the continuous sensing scenario presented in Fig. 7, we varied the frequency of the voluntary data publications made by data collectors to the platform, as follows: each 30 secs, each 1 minute, each 5 minutes, and each 10 minutes. The following metrics were measured: response time, network load, and traffic estimation error.
	Network load	Size of packets exchanged for the end-to-end interaction (between voluntary publication of data and traffic condition response)	
	Traffic estimation error %	Calculated using equation 6	
System Data Frequency Based Testing- On Demand Sensing Approach	Response time	Time from when sensing request (msg. 1 in fig. 6) is sent until sensing response (msg. 7 in fig. 6) is received.	Using the on-demand sensing scenario presented in Fig. 6, we varied the number of received sensing requests/hour by the platform, as follows: 10 requests/hr., 100 requests/hr., 500 requests/hr., 1000 requests/hr., and 10,000 requests/hr. The following metrics were measured: response time, network load, and traffic estimation error.
	Network load	Size of packets exchanged for the end-to-end interaction (between sensing request and sensing response)	
	Traffic estimation error %	Calculated using equation 6	

Participation % Based Testing	Response time	Time from when sensing request (msg. 1 in fig. 6) is sent until sensing response (msg. 7 in fig. 6) is received.	Using the on-demand sensing scenario presented in Fig. 6, we varied the % of cars participating in the sensing activity and to see the impact on the accuracy of the results. The % of targeted cars was varied from 100% of cars (continuous sensing case), to 70%, to 50%, to 30%, to 10%. The following metrics were measured: response time, network load, and traffic estimation error.
	Network load	Size of packets exchanged for the end-to-end interaction (between sensing request and sensing response)	
	Traffic estimation error %	Calculated using equation 6	
Quality of Sensed In- formation Testing	Traffic estimation error	Calculated using equation 6	Using the on-demand sensing scenario presented in Fig. 6, we varied the % of cars participating in the sensing activity (from 100% to 10%) as well as the matching criteria used (using 6, 4, 3, 2, and 1 matching criterion) in order to study the impact of the selection criteria on the Quality of sensed information & traffic estimation accuracy. The traffic estimation error was measured in that case.

3.5 Experimental Results and Discussion

3.5.1 Performance Evaluation Strategy and Metrics

The objectives of the experiments we conducted are to: (1) assess the performance of the two main algorithms implemented by the on-demand sensing platform (i.e. the matching and the traffic estimation algorithms); (2) evaluate the overall system performance including all the communications and processing overhead; and (3) compare the performance of the on-demand and continuous sensing approaches, using the two test scenarios presented in figures 6 and 7.

To achieve those goals, a number of testing approaches and performance metrics were used, as summarized in table 3.4.2. The detailed analysis of the conducted tests will be presented in the coming sections.

Figure 8: Simulated roads' topology

3.5.2 Algorithms' Performance Evaluation

3.5.2.1 Traffic estimation algorithm

Figure 9: Traffic estimation results

Figure 9 depicts the performance of our traffic estimation algorithm, when applied to four types of roads: a free flowing road, a moderately congested road, a congested road, and a highly congested road. For each scenario, we calculated the estimated mean speed,

the ground truth for the road, and the traffic estimation error, as shown in the figure. By analyzing the obtained results, we notice that the mean speed estimation method yields more accurate results in the free flowing roads than in the more congested road, with an estimated mean speed of 39 Km/hr. on a road with a ground truth of 30.26 Km/hr (for the free flowing case), vs. an estimated mean speed of 6.39 Km/hr. on a highly congested road with a ground truth of 3.16 Km/hr. Another observation is that the mean speed method resulted in speed over-estimation in both congested and uncongested conditions. In absolute vehicular speed terms, the obtained traffic estimation results are very good since we are more interested in the traffic status (i.e. free flowing, moderately congested, congested, and highly congested) rather than the actual speed on the road. Thus, since the estimated traffic mean speed values were close to the ground truth values on the tested roads, the correct traffic condition was inferred in the four tested scenarios.

3.5.2.2 Matching algorithm

Figure 10(a) shows the performance of the multi-criteria matching algorithm, when all six selection criteria (i.e. availability, proximity, data collection capability, accuracy, battery level, and reputation) are used for participants' selection. In this experiment, the number of cars on the road, which were processed during the matching varied from 50 cars to 1000 cars. As expected, when the number of cars available on the road increased (i.e. the size of the dataset increased), the number of targeted collectors matching the selection criteria increased. For instance, when the number of cars on the road is 50, the number of selected data collectors is 4, while when there were 1000 cars on the road, the number of targeted collectors rose to 80. Examining the performance of the 6-criteria matching algorithm, we observe that the algorithm yielded good results, by selecting 4

Figure 10: Performance for six/four/three/two criteria of matching algorithm

out of 4 eligible cars for a dataset of 50 cars (i.e. matching error percentage of 0%), and 10 out of 11 eligible cars for a dataset of 100 cars (i.e. matching error percentage of 10%). We notice that the matching error % increases with an increase of the size of the cars datasets processed. For instance, when the dataset consisted of 800 cars, the number of selected cars was 37 cars out of 52 eligible cars (i.e. a matching error percentage of 29%). As for the matching algorithm's response time, it varied between 220 ms in the case of 50 processed cars to 296 ms when 1000 cars were processed, which is an acceptable performance that would bear minimum impact on the end-to-end system response time. It should be noted that the other variants of the multi-criteria matching algorithm exhibit a similar performance from response time and matching error %, as shown in figure 10(b), (c) and (d).

3.5.2.3 System's Performance Evaluation

- *Load Testing:*

Figure 11: Load testing results for on-demand sensing platform

In order to evaluate the behavior of the on-demand sensing system under variable loading conditions, we conducted some load tests using the test setup shown in figure 6. Figure 11 shows the obtained load testing results.

As shown in figure 11, the on-demand sensing system shows a polynomial (cubic) growth pattern in terms of response time, which ranged from 14.3 sec for 1 sensing request to 22.95 sec for a 2000 sensing requests. It worth to mention that the response time can be reduced significantly when using computers with better performance setup since most of it is spent on algorithmic computation. This polynomial response time growth pattern can be attributed to four time consuming steps related to on-demand sensing, namely: the

multi-criteria participants' selection process; the waiting time required to receive sensed data from the targeted participants; the concurrent access to platform's DB for storage of different pieces of sensed data; the traffic estimation process and generation of traffic reports.

As for the generated network load, it showed a polynomial (quadratic) growth pattern with values ranging from 120 KB for 1 sensing request to 21689 KB for 2000 sensing requests. The network load's growth pattern can be explained by the fact that the more sensing requests are received, the more data collectors are targeted which multiplies the number of messages exchanged through the system

- *Data-Frequency Based Testing in On-Demand vs Continuous Sensing:*

In order to compare the on-demand and continuous sensing approaches, we conducted data frequency based testing in which we varied the sensing frequency (i.e. the number of sensing requests received per hour by the on-demand sensing platform and the number of voluntary publications made in continuous sensing mode) and measured the response time and network load generated in both cases. The presented results in Figures 12 and 13 illustrate clearly the benefits of the proposed on-demand approach in terms of traffic overhead and network load compared to the continuous, while maintaining very close traffic estimation accuracy in both of them.

Figure 12 shows the data frequency based testing results for the on-demand sensing approach in terms of network load, response time, and traffic estimation accuracy (error %) as a function of the average number of received sensing requests/hour. As shown in the figure, the on-demand sensing system shows a logarithmic growth pattern in terms

Figure 12: Data frequency based testing results for on-demand sensing approach

of response time, which increases from 15.264 sec to 30.165 sec as the average number of received sensing requests per hour increases from 10 to 10000. This growth pattern can be explained by the fact that the more requests are received per hour, the more fresh data is available in the platform, which can be reused to answer subsequent requests without the need to resort to data collectors. Furthermore, in some cases, the current traffic status reports may be already available in the system due to many requests in the same area, which will decrease the response time to the new data consumers requesting traffic conditions in the same area. On the other hand, the system's network load exhibits a polynomial (quadratic) trend line, ranging from 340 KB for an average of 10 sensing requests received/hour up to 3568 KB for an average of 10000 sensing request received/hour. This polynomial increase is attributed to the additional number of data collectors required

for new requests, thus generating additional traffic load. As for the traffic estimation accuracy, we notice that as the average number of sensing received by hour increases, the traffic estimation error % decreases, dropping from 17.82% estimation error for 10 sensing requests received/hour to 2.9% estimation error for 10000 sensing requests received/hour. This can be explained by the fact that the more requests are received, the more data points are collected about a certain area, and the more accurate the traffic estimation results will be.

Figure 13: Data frequency based testing results for continuous sensing approach

On the other hand, Figure 13 shows the data frequency based testing results for the continuous sensing approach in terms of network load, response time, and traffic estimation accuracy (error %) as a function of the voluntary data publication frequency. Similar to the on-demand sensing case, in the continuous sensing case, we notice that the traffic

estimation error decreases with the increase of the voluntary data publication frequency, dropping from an estimation error of 17.7% for data voluntarily published each 10 minutes to an estimation error of 5.9% for data published each 30 seconds. On the other hand, the network load follows a polynomial (quadratic) growth curve, which is expected with the increase in the number of data publication messages associated with an increased publication frequency (i.e. from each 10 minutes to each 30 seconds) . Finally, we notice that the response time remains constant with respect to the voluntary data publication frequency. This is due to the fact that when a traffic condition request is received by the platform, it is using the data previously published in the system to estimate the traffic and send the final response. Therefore, the data publication frequency bears no effect on the response time in the continuous sensing case.

- *Participation Percentage Based Testing:*

The continuous sensing approach can be considered as a special case of on-demand sensing approach, in which data is acquired on a regular basis from a 100% of the cars, instead of occasionally from some of the cars. In order to test the impact of the percentage of cars participating in the sensing activity on the accuracy of the traffic estimation results, we carried a test in which the % of participating cars is varied from 10% to 100% and measured the response time, network load, and traffic estimation error. Figure 14 depicts the obtained results.

As observed, both the network load and response time increase in a linear fashion with the increase in the % of cars participating in the sensing activity. For instance, the network load and response time respectively achieved for 10% of cars targeted are 110 KB and

Figure 14: Participation percentage based testing results

3 seconds. For a 100% of cars targeted (i.e. the continuous sensing case), the network load and response time increased to 1760 KB and 15.5 seconds. This is an expected result as more participation results in more message exchange (i.e. higher network load) and more time to process those messages (i.e. higher response time). On the other hand, we notice that the traffic estimation error decreases in a logarithmic fashion, with the increase in the % of participating cars. This is due to the fact that the more cars are targeted, the more data points are collected about a certain area, and the more accurate is the traffic estimation result. It should however be mentioned that there is a compromise between the accuracy of the traffic estimation result needed and the system's performance in terms of network load and response time. In fact, a higher traffic estimation accuracy will be associated with a poorer system performance in terms of network load and response time.

For instance, with a 100 % of cars targeted, we obtain the lowest traffic estimation error (i.e. 8.3%) along with the highest network load (i.e. 1760 KB) and the highest response time (i.e. 15.5 seconds). Decreasing the percentage of car participation to 50% results in penalty of 4% of additional traffic estimation error, but an improvement of 50.9% in terms of network load and an improvement of 46.5% in terms of response time. In the case of 30% participation rate, the additional traffic estimation error accrued is 10%, while the improvement in terms of network load is 73.8% and the improvement in terms of response time is 60.3%. Moreover, it is worth to mention about the high accuracy achieved by the proposed on-demand approach, where in all the cases the traffic estimation error is acceptable in order to determine the traffic status of the road, especially 30% and above, where the error rate starts to be similar to the continuous approach.

- *Impact of Selection Criteria on Quality of Sensed Information and Traffic Estimation*

Accuracy:

In order to evaluate the impact of the participants' selection criteria on the traffic estimation accuracy, we varied both the % of cars targeted for a sensing activity as well as the # of criteria used for participants' selection from the ones targeted. Figure 15 depicts the obtained results.

As expected, for the 5 sets of matching/selection criteria used (i.e. 6 matching criteria, 4 matching criteria, 3 matching criteria, 2 matching criteria, and 1 matching criterion), increasing the % of cars targeted for sensing has a positive impact on the traffic estimation accuracy. Furthermore, when the same % of cars are targeted and the different variants of the matching approach are compared, the more selection criteria we use, the more accurate

Figure 15: Impact of Selection Criteria on Traffic Estimation Accuracy

is the traffic estimation result. As shown in the figure, the 6-criteria matching approach (the blue curve) outperforms all other approaches (i.e. 4 criteria, 3 criteria, 2 criteria, 1 criteria) for all % of participating cars used. This is due to the fact that for the same % of cars targeted, the 6-selection criteria approach selects the best candidates yielding the highest quality records satisfying multiple quality of information criteria. Although the other approaches select the same number of candidates in each test scenario, the selected candidates provide lower quality information since some of the quality criteria are not considered in the selection process, thus yielding less accurate traffic estimation results.

It is very important to mention that even with less % of targeted cars, the selection approaches with more criteria outperform those with less selection criteria targeting a higher % of cars in some cases. As an example illustrated in the blue dotted area in Figure 15, the

6-selection criteria approach achieves a traffic estimation error percentage of 29.015% with only 10% of targeted cars, which is a lower traffic estimation error than the ones achieved by the 2-criteria approach and 1-criteria approach targeting 30% and 50% of the cars (yielding traffic estimation errors ranging between 30.29% and 38.84%). The same applies when comparing the 6-criteria approach targeting 30% of cars, to all other variants targeting 50%, 70%, and even 100% of cars (see green dotted area in the figure). This implies that using the 6 criteria approach and targeting 30% cars as candidates' yields more accurate results than targeting 100% of cars with only 4 selection criteria. We can therefore conclude that there exists a tradeoff between the number of selection criteria (i.e. the complexity of the selection approach) and the accuracy of the traffic estimation results obtained. Therefore, when more contextual information is available and multiple selection criteria can be used for participants' selection, a small % of participating cars can be targeted while achieving high traffic estimation accuracy. On the other hand, in the case of lack of availability of contextual information and the inability to use multiple selection criteria, a larger % of participating cars need to be targeted to compensate for the lower quality data records and maintain good traffic estimation accuracy. Based on the tests we conducted and the results obtained, we can conclude that using 6 matching criteria and 30% of targeted cars achieves the best trade-off in the test scenario and environment we used.

3.6 Conclusion

We presented in this chapter an On-Demand approach for road traffic monitoring. The proposed solution is based on a multi-criteria data collectors selection that considers the

participants presence in the region of interest, their availability and capability to participate in the sensing activities, their phones battery level, the accuracy of the sensed data they provide and the users reputation. The approach also adopts a mean speed based traffic estimation to get the real-time roads condition and the communication among the different system components is conducted through RESTful web service communication interfaces. Finally, we discuss the results of the performance analysis done to show the usefulness of our proposition.

Chapter 4

Traffic Condition Estimation based on On-Demand Sensing with the support of Mobile Infrastructure

4.1 Introduction

In the previous chapter, we addressed the problem of estimating the mean speed by assuming that each data collector periodically shares its sensed data with the server. However, gathering user status including geographic location every two minutes is considered as a limitation to our full on-demand goal within ITS context since it entails persistent sampling of sensed traffic data. Moreover, a data consumer will be provided with the value of the mean speed (in km/h) whenever he requests the traffic status of a road segment. This information may not satisfy the requester as it doesn't reflect a complete picture about the traffic condition in that area. To address the aforementioned limitations, we propose in

this chapter an enhanced sensing approach supporting fully on-demand strategy in order to classify the traffic statuses. The new approach embeds new models for both Matching and Traffic Estimation. First, we elaborate a complementary method to infer the identity of vehicles located on a particular area through the use of cellular networks. Such method would prevent the users from publishing continuously their sensed data by collecting, at any request time, the targeted nodes aggregated to cell towers that cover an area of interest. Second, we propose a combined classification approach to infer traffic condition based on density and mean speed while adopting Kerner's three phase traffic theory [11] that divides the traffic into three categories: Jam, moderate, and free flow.

The rest of the chapter is organized as follows. Section 4.2 represents the approach overview. We elaborate in section 4.3 the vehicle selection model through mobile cell towers and in section 4.4 the classification based model for traffic condition. The performance evaluation is presented in section 4.5 and the chapter is finally concluded in section 4.6

4.2 Approach Overview

Both matching/participant selection and traffic estimation are considered the main models in our vehicular sensing platform presented in figure 1 of chapter3-section 3.2. Figure 16 illustrates the architecture of our proposed approach including the new elaborated models for participant selection and traffic estimation. When the server receives sensing request from data consumer, it matches the request with the cell towers located in the specified area in order to collect all nodes connected to them. A first sensing request is then sent to each node from the matched set requesting its specific location. All sensed data coming from

Figure 16: High Level of Fully On-Demand Vehicular Sensing System Architecture

in-vehicles sensors and found on the targeted road are selected to form the most suitable set of collectors. Since the traffic estimation model requires two sensed data from each sensor on road to predict the traffic condition, a second sensing request is therefore sent after 10 seconds to the elected set of nodes. Once all the nodes reply, the server runs the algorithm implementing the traffic estimation model. Our proposed model combines the density and mean speed characteristics to best reflect the road status and the resulting traffic condition is symbolized either by red for Jam, yellow for synchronized, or by green for free flowing. Table 5 presents the notations of the formulas notations used in both models, which we present their technical details in the following sections.

Table 5: Formulas Notations

Variable	Description
ST	Set of Tower
SS	Set of Sensor
R	The desired road from which the traffic condition is requested
R'	An adjacent road heading from R
$S_{id}^{T_i}$	Sensor S with ID = id collected from Tower T_i
$S_j.avail$	The availability of S
$pos_{t_i}^{S_j}$	Position of S_j at t_i
$S_j.sType$	Origin of S_j 's sensed data
V_{max}	Maximum speed that could be reached on a road
TC	Traffic Condition
FF	Free Flowing
MC	Moderate Congestion
TJ	Traffic Jam

4.3 Vehicle selection Model through mobile Cell Towers

The proposed matching model relies on several criteria to match the consumer request with the appropriate list of collectors in order to get the data of their geographic locations. Once the data consumer sends sensing request to the platform requesting the traffic condition on a specific road, the server launches the matching module.

The proposed model supports fully on-demand approach, without requiring any continuous or previous sensed data from the nodes. Its main criteria are both the geographic location inferred from cell towers data and availability of the targeted collectors, in addition to their phones battery level, reputation, sensing capabilities, and the accuracy of their

sent data. First, the cell towers that cover the entire road, from which the traffic condition is requested, are identified based on a map topology and predefined towers locations. The number of towers chosen is defined by:

$$ST : \sum_{i=1}^n T_i \quad | \quad T_i \text{ covers } R \quad (1)$$

where ST contains the set of all the towers T_i surrounding the road R through their coverage area

Figure 17: Representation of the road topology and cellular towers locations on the case study area

The cell towers have different sizes. The large ones, which are usually found on highways, are called macrocells and offer wide area coverage. Over a smaller area, microcells are used to cover urban and suburban cells. Moreover, picocells are employed for even smaller coverage area such as buildings, campuses, and airports [34]. In the proposed approach, we deal with microcells that cover around one mile in diameter since our study

addresses the urban roads. Fig 17 illustrates the road topology used in the simulated scenarios including the cell towers with hexagonal shape covering the case study area. If a data consumer requests the status of the road x presented in the figure, towers T2 and T3 are picked to capture the vehicles movement. Then, all the nodes IDs connected to the chosen towers are requested from the mobile providers. The collected set of nodes is defined by:

$$SS_{init} : \bigcup_{i=1}^n S_{id}^{T_i} \quad (2)$$

where SS_{init} stands for the initial set of sensors that includes the union of their IDs gathered from the different requested Towers.

Afterwards, sensing request is sent to each node in the SS_{init} participating in the sensing activities to get its geographic location at the request time t_1 , along with the origin of the data sent (i.e. pedestrian or vehicle). The current mobile devices provide new feature to recognize whether sensed data comes from a pedestrian or in-vehicle device. Therefore, we filter the obtained set to keep only the on-road vehicular sensors, after eliminating the pedestrians' nodes. Hence, the filtered set of sensors after receiving the nodes responses becomes as follows:

$$SS_{filtered} : \sum_{j=1}^n (S_j.avail == true) \wedge (pos_{t_1}^{s_j} \in R) \wedge (S_j.sType == Veh) \quad (3)$$

where $SS_{filtered}$ holds all the available nodes willing to participating in the data collection activities, located on the road R at t_1 , and having vehicular (Veh) sensor type.

To obtain the two required sensed data from the collectors to estimate the road condition, a second sensing request is sent to the set $SS_{filtered}$ after 10 seconds of t_1 to get the new nodes positions as responses. The final set of sensors SS_{final} identified after receiving $SS_{filtered}$ responses is represented by equation (4)

$$SS_{final} : \sum_{j=1}^n (S_j.avail == true) \wedge [pos_{t_2}^{s_j} \in R \vee pos_{t_2}^{s_j} \in R'] \wedge (S_j.sType == Veh) \quad (4)$$

where SS_{final} holds all the available nodes, located on R or R' (the adjacent road heading from R) at t_2 , and having vehicular (Veh) sensor type.

We developed the algorithm of the aforementioned model within the matching module of the sensing platform. Algorithm 2 illustrates all the steps of the matching including the interactions with the other modules of the platform.

4.4 Classification-based Model for Traffic Conditions

In this section, we present our proposed model for traffic estimation that aims to classify the different traffic flow conditions based on Kerner's theory [11]. Density, mean speed and flow [35] are the three main characteristics used to evaluate the traffic stream in macroscopic traffic flow model. Some proposed approaches [36], [37] focus on two characteristics to estimate the traffic condition, while others [18], [30] use only one. In our proposed model, we base our estimation on an approach combining both density and mean speed. Moreover, we adopt Kerner's theory that divides the traffic into three categories: 1) Traffic

Algorithm 2 - Matching Algorithm through Cellular Tower

- 1: **Input:** Map Topology + Towers locations
 - 2: **Output:** Set of targeted cars SS_{final} located in the specified destination
 - 3: t_1 = time of consumer request
 - 4: get the set of towers ST covering the targeted road R
 - 5: Construct a list $S_{init} : \emptyset$ for the initial set of sensors
 - 6: **for** each T_i in ST **do**
 - 7: send request to T_i 's provider, and get its nodes IDs S_{id}
 - 8: $SS_{init} = SS_{init} \cup S_{id}^{T_i}$
 - 9: **for** each sensor S_j in SS_{init} **do**
 - 10: send sensing request to S_j , and get its position pos^{S_j} at t_1
 - 11: Construct a list $S_{filtered} : \emptyset$ for the set of sensors
 - 12: **for** each sensor S_j in SS_{init} **do**
 - 13: **if** $S_j == \text{available}$ **then**
 - 14: **if** $pos_{t_i}^{S_j} == \text{onRoad} \ \&\& \ S_j \text{ in } V$ **then**
 - 15: add S_j to $SS_{filtered}$
 - 16: $t_2 = t_1 + 10 \text{ secs.}$ / t_1 = time of consumer request
 - 17: **for** each sensor S_j in $SS_{filtered}$ **do**
 - 18: send sensing request to S_j , and get its position pos^{S_j} at t_2
 - 19: Construct a list $S_{final} : \emptyset$ for the set of sensors
 - 20: **for** each sensor S_j in $SS_{filtered}$ **do**
 - 21: **if** $S_j == \text{available}$ **then**
 - 22: **if** $pos_{t_i}^{S_j} == \text{onRoad } R \parallel pos_{t_i}^{S_j} == \text{onRoad } R'$ **then** $\&\& \ S_j \text{ in } V$
 - 23: add S_j to SS_{final}
-

Jam describing a wide traffic, 2) Moderate congestion, also known as Synchronized Flow, showing no significant stoppage of vehicles, and 3) Free Flow reflecting continuous traffic flow with no congestion. Each of the density and mean speed classifies the traffic status into one of these categories. In case of conflicting results, a percentage-based resolution strategy is proposed to provide the final decision, which is sent to the consumer using red, yellow, or green color. In the sequel, we provide the technical details of our approach.

4.4.1 Density based Estimation

The density based estimation represents the number of vehicles occupying the segment of the requested road as denoted in the following:

$$K = \frac{N}{L} \quad (5)$$

where N is equal to the number of vehicles on R , and L is the length of R (in miles). N is obtained from equation (5) when the set of $SS_{filtered}$ shares their location at t_1 . Table 6 shows how the different traffic conditions are deduced when k varies between zero and the maximum number of vehicles over one mile. [38]

Table 6: Relations between traffic condition and density

Traffic Condition	k	k
	(vehicles/mile)	(vehicles/305meters)
Free Flowing	$0 \leq k \leq 30$	$0 \leq k \leq 5$
Moderate Congestion	$30 < k \leq 160$	$5 < k \leq 30$
Traffic Jam	$160 < k \leq 233$	$30 < k \leq 44$

By analyzing the flow condition on road x in Fig. 17, we can find that the length of the road is 305 meters, and therefore the relation between the traffic condition and the number of vehicles is calculated and provided in table 6.

4.4.2 Mean Speed based Estimation

To calculate the mean speed at a given time, we need to obtain data collected from a set of sensors located in a specific area of interest. More precisely, a pair of latitude and longitude coordinates data consecutively sampled from each sensor is needed. Consequently, the nodes geographic location in both sets $SS_{filtered}$ (equation 3) and SS_{final} (equation 4) are used to estimate the real mean speed. To start with, the distance of road traveled by sensor S_j is characterized by

$$r_j\left(pos_{t_1}^{S_j}, pos_{t_2}^{'S_j}\right) \quad (6)$$

where the position of S_j at t_1 ($pos_{t_1}^{S_j}$) is always on R, however $pos_{t_2}^{S_j}$ could either be R or R' that represents any adjacent road heading from R. Therefore, we adopt $pos_{t_2}^{'S_j}$ as S_j position at t_2 which denotes the intersection of the end/exit road if $pos_{t_2}^{S_j}$ is on R', or the $pos_{t_2}^{S_j}$ if it is on R.

Moreover, the average speed v_j of each sensor is represented by:

$$v_j = \frac{\left(pos_{t_1}^{S_j}, pos_{t_2}^{'S_j}\right)}{(t_1, t_2)} \quad (7)$$

The formula used for the mean speed is calculated based on equations 6 and 7 and can be

defined as

$$v_{mean} = \frac{\sum_{s_j \in SS_{final}} \left[v_j \times r_j \left(pos_{t_1}^{S_j}, pos_{t_2}^{S_j} \right) \right]}{\sum_{s_j \in SS_{final}} r_j \left(pos_{t_1}^{S_j}, pos_{t_2}^{S_j} \right)} \quad (8)$$

The classification of the mean speed v_{mean} into the three levels is estimated based on the thresholds [39] illustrated in table 7, where V_{max} represents the maximum allowed speed on a specified road.

Table 7: Relations between traffic condition and mean speed

Traffic Condition	v_{mean} (km/h)
Free Flowing	$13 \leq v_{mean} \leq v_{max}$
Moderate Congestion	$7 \leq v_{max} < 13$
Traffic Jam	$0 \leq v_{max} < 7$

4.4.3 Rule-Based Inferred Traffic Condition

As previously mentioned, the final traffic condition category is deduced from both density and mean speed models after combining their estimated results. If the two resulting conditions match, then a straightforward decision is inferred and sent to the consumer. In this context, the final congestion level is red, yellow or green when both density and mean speed show traffic jam, moderate congestion, or free flowing condition respectively. Otherwise, it may happen that each model classifies the road condition into different congestion level. When such a conflict occurs, the estimated levels could not be averaged out since we adopt only two criteria. To address the conflicting problem between the two models, we propose the following strategy which ensures the validity of the final result:

"The further the value of a criteria is away from its decision boundaries, the more accurate its classification estimation will be"

For instance, let's take the example where k is equal to 35 vehicles/mile and v_{mean} is equal to 4 km/h. Therefore, the density-based model estimates a Moderate Congestion condition, while the mean speed based model reports a Traffic Jam. Thereby, we calculate how each variable is away from the boundary of the conflicting level, and hence we consider the final classification of the one having higher value. In this case, between k that falls between 30 and 160 and how close from 30 is and v_{mean} that falls between 0 and 7 and how close from 7 is, we can say that k is closer to the considered boundary and the final traffic condition is hence the one of the mean speed.

To realize formally the semantics in case of conflicting, we elaborated inference rules based on deductive logic to decide on the final road status and classify it as Free Flow (FF), Moderate Congestion (MC), and Traffic Jam (TJ). Inference rules usually have standard structure, where the conclusion is presented below a horizontal line and a list of premisses listed above the line. [40]

The final traffic condition (TC) is represented as follows:

$$\frac{(\text{premise}_1) \text{ op } (\text{premise}_2) \text{ op } \dots \text{ op } (\text{premise}_n)}{\langle \text{TC, Final} \rangle \xrightarrow[\text{classify}]{} \text{FF/MC/TJ}}$$

where op is either $\vee$ to represent the logical operator "and", or $\wedge$ to represent the logical operator "or".

Premises 1 and 2 denote the traffic conditions (TC) estimated from density and mean speed

models respectively.

$$\langle \text{TC, Density} \rangle \xrightarrow{\text{classify}} \text{FF/MC/TJ} \quad (\text{premise 1})$$

$$\langle \text{TC, Mean Speed} \rangle \xrightarrow{\text{classify}} \text{FF/MC/TJ} \quad (\text{premise 2})$$

where TC is classified to FF, MC or TJ

The other premises are combinations of rules-based of the calculated ranges (v_{range} and k_{range}) needed in case of conflict in premise 1 and premise 2 classification.

In the sequel, we present the details of calculating v_{range} and k_{range} , in addition to the inference rules for conflicting classification.

4.4.3.1 Mean Speed and Density Range Calculation

This strategy: "**The further the value of a criteria is away from its decision boundaries, the more accurate its classification estimation will be**" is yet opted since we have confidence in the classification of the variable (density/mean speed) that has a value close to the middle of its boundaries.

To realize the proposed strategy, we first adopt the boundaries notations presented in table 8.

Table 8: Boundaries notations for the density and mean speeds

Traffic Condition	k (vehicles/mile)	v_{mean} (km/h)
Free Flowing	$Bk1 \leq k \leq Bk2$	$Bv1 \leq k \leq Bv2$
Moderate Congestion	$Bk2 < k \leq Bk3$	$Bv2 \leq k < Bv3$
Traffic Jam	$Bk3 < k \leq Bk4$	$Bv3 \leq k < Bv4$

Second, we get the two boundaries Bk_i and Bk_j where the density goes between, and the two corresponding ones Bv_i and Bv_x for the mean speed. Then, for the density k (computed in equation 5) and mean speed v (computed in equation 8), we calculate how far away from their boundaries are as percentage value following equations 9 and 10 respectively

$$k_{range} = \frac{|k - Bk_i|}{|Bk_j - Bk_i|} \times 100 \quad (9)$$

$$v_{range} = \frac{|v - Bv_i|}{|Bv_x - Bv_i|} \times 100 \quad (10)$$

Eventually, the final traffic condition tends to consider the classification of the variable that has higher range. Consider the following cases which summarize all combinations of the density and mean speed leading to conflicting problem:

Traffic Condition	k (vehicles/mile)	v_{mean} (km/h)
Free Flowing	$0 \leq k \leq 30$	$13 \leq v_{mean} \leq V_{max}$
Moderate Congestion	$30 < k \leq 160$	$7 \leq v_{mean} < 13$
Traffic Jam	$160 < k \leq 233$	$0 \leq v_{mean} < 7$

- **Case 1:** when density estimates free flowing condition, while mean speed estimates moderate congestion. The conflicting levels are Free Flowing (FF) and Moderate Congestion (MC). Thus, the density range is represented by:

$$k_{range: MC \rightarrow FF} = \frac{|K - Bk_2|}{|Bk_1 - Bk_2|} \times 100$$

where density tends to change the traffic status from MC to FF by calculating the distance from k value to Bk_2

And the mean speed range is represented by:

$$v_{\text{range: FF} \rightarrow \text{MC}} = \frac{|v - Bv_2|}{|Bv_3 - Bv_2|} \times 100$$

where mean speed tends to change the traffic status from FF to MC by calculating the distance from v_{mean} to Bv_2

- **Case 2:** when density estimates moderate congestion condition, while mean speed estimates free flowing congestion. The conflicting levels are Moderate Congestion and Free Flowing. Thus, the density range is represented by:

$$k_{\text{range: FF} \rightarrow \text{MC}} = \frac{|K - Bk_2|}{|Bk_3 - Bk_2|} \times 100$$

where density tends to change the traffic status from FF to MC by calculating the distance from k value to Bk_2

And the mean speed range is represented by:

$$v_{\text{range: MC} \rightarrow \text{FF}} = \frac{|v - Bv_2|}{|Bv_1 - Bv_2|} \times 100$$

where mean speed tends to change the traffic status from MC to FF by calculating the distance from v_{mean} to Bv_2

- **Case 3:** when density estimates moderate congestion condition, while mean speed

estimates traffic jam congestion. The conflicting levels are Moderate Congestion and Traffic Jam (TJ). Thus, the density range is represented by:

$$k_{\text{range: TJ} \rightarrow \text{MC}} = \frac{|K - Bk_3|}{|Bk_2 - Bk_3|} \times 100$$

where density tends to change the traffic status from TJ to MC by calculating the distance from k value to Bk_3

And the mean speed range is represented by:

$$v_{\text{range: MC} \rightarrow \text{TJ}} = \frac{|v - Bv_3|}{|Bv_4 - Bv_3|} \times 100$$

where mean speed tends to change the traffic status from MC to TJ by calculating the distance from v_{mean} to Bv_3

- **Case 4:** when density estimates traffic jam condition, while mean speed estimates moderate congestion. The conflicting levels are Traffic Jam and Moderate Congestion. Thus, the density range is represented by:

$$k_{\text{range: MC} \rightarrow \text{TJ}} = \frac{|K - Bk_3|}{|Bk_4 - Bk_3|} \times 100$$

where density tends to change the traffic status from MC to TJ by calculating the distance from k value to Bk_3

And the mean speed range is represented by:

$$v_{\text{range: TJ} \rightarrow \text{MC}} = \frac{|v - Bv_3|}{|Bv_2 - Bv_3|} \times 100$$

where mean speed tends to change the traffic status from TJ to MC by calculating the distance from v_{mean} to Bv_3

4.4.3.2 Inference Rules for Traffic Condition Classification

In this section, we present the inference rules of the final traffic condition classifications illustrated in Figures 18, 19 and 20.

$$\begin{array}{c}
 \left(\left(\langle \text{TC, density} \rangle \xrightarrow{\text{classify}} \text{FF} \right) \wedge \left(\langle \text{TC, MeanSpeed} \rangle \xrightarrow{\text{classify}} \text{FF} \right) \right) \\
 \vee \left(\left(\langle \text{TC, density} \rangle \xrightarrow{\text{classify}} \text{FF} \right) \wedge \left(\langle \text{TC, MeanSpeed} \rangle \xrightarrow{\text{classify}} \text{MC} \right) \wedge (k_{\text{range: MC} \rightarrow \text{FF}} > v_{\text{range: FF} \rightarrow \text{MC}}) \right) \\
 \vee \left(\left(\langle \text{TC, density} \rangle \xrightarrow{\text{classify}} \text{MC} \right) \wedge \left(\langle \text{TC, MeanSpeed} \rangle \xrightarrow{\text{classify}} \text{FF} \right) \wedge (v_{\text{range: MC} \rightarrow \text{FF}} > k_{\text{range: FF} \rightarrow \text{MC}}) \right) \\
 \hline
 \langle \text{TC, final} \rangle \xrightarrow{\text{classify}} \text{FF}
 \end{array}$$

Figure 18: Final traffic condition classified to FF

In figure 18, the final TC is classified to FF if the TC inferred from both density and mean speed is FF, or the density-based TC estimation is FF while the mean speed estimation is MC and k_{range} is greater than v_{range} , or the density estimation is MC while the mean speed estimation is FF and v_{range} is greater than k_{range} .

In figure 19, the final TC is classified to MC if one of the following five cases occurs:

- Both density and mean speed estimate the TC as MC

$$\begin{array}{c}
\left(\langle \text{TC, density} \rangle \xrightarrow{\text{classify}} \text{MC} \right) \wedge \left(\langle \text{TC, MeanSpeed} \rangle \xrightarrow{\text{classify}} \text{MC} \right) \\
\vee \left(\left(\langle \text{TC, density} \rangle \xrightarrow{\text{classify}} \text{FF} \right) \wedge \left(\langle \text{TC, MeanSpeed} \rangle \xrightarrow{\text{classify}} \text{MC} \right) \wedge (v_{\text{range: FF} \rightarrow \text{MC}} > k_{\text{range: MC} \rightarrow \text{FF}}) \right) \\
\vee \left(\left(\langle \text{TC, density} \rangle \xrightarrow{\text{classify}} \text{MC} \right) \wedge \left(\langle \text{TC, MeanSpeed} \rangle \xrightarrow{\text{classify}} \text{FF} \right) \wedge (k_{\text{range: FF} \rightarrow \text{MC}} > v_{\text{range: MC} \rightarrow \text{FF}}) \right) \\
\vee \left(\left(\langle \text{TC, density} \rangle \xrightarrow{\text{classify}} \text{MC} \right) \wedge \left(\langle \text{TC, MeanSpeed} \rangle \xrightarrow{\text{classify}} \text{TJ} \right) \wedge (k_{\text{range: TJ} \rightarrow \text{MC}} > v_{\text{range: MC} \rightarrow \text{TJ}}) \right) \\
\vee \left(\left(\langle \text{TC, density} \rangle \xrightarrow{\text{classify}} \text{TJ} \right) \wedge \left(\langle \text{TC, MeanSpeed} \rangle \xrightarrow{\text{classify}} \text{MC} \right) \wedge (v_{\text{range: TJ} \rightarrow \text{MC}} > k_{\text{range: MC} \rightarrow \text{TJ}}) \right) \\
\hline
\langle \text{TC, final} \rangle \xrightarrow{\text{classify}} \text{MC}
\end{array}$$

Figure 19: Final traffic condition classified to MC

- Density based estimation is FF, while mean speed based estimation is MC and $v_{\text{range}} > k_{\text{range}}$
- Density based estimation is MC, while mean speed based estimation is FF and $k_{\text{range}} > v_{\text{range}}$
- Density based estimation is MC, while mean speed based estimation is TJ and $k_{\text{range}} > v_{\text{range}}$
- Density based estimation is TJ, while mean speed based estimation is MC and $v_{\text{range}} > k_{\text{range}}$

In figure 20, the final TC is classified to TJ if the TC inferred from both density and mean speed is TJ, or the density-based TC estimation is TJ while the mean speed estimation is MC and k_{range} is greater than v_{range} , or the density estimation is MC while the mean speed estimation is TJ and v_{range} is greater than k_{range} .

$$\begin{array}{c}
\left(\left(\langle \text{TC}, \text{density} \rangle \xrightarrow{\text{classify}} \text{TJ} \right) \wedge \left(\langle \text{TC}, \text{MeanSpeed} \rangle \xrightarrow{\text{classify}} \text{TJ} \right) \right) \\
\vee \left(\left(\langle \text{TC}, \text{density} \rangle \xrightarrow{\text{classify}} \text{TJ} \right) \wedge \left(\langle \text{TC}, \text{MeanSpeed} \rangle \xrightarrow{\text{classify}} \text{MC} \right) \wedge (k_{\text{range: MC} \rightarrow \text{TJ}} > v_{\text{range: TJ} \rightarrow \text{MC}}) \right) \\
\vee \left(\left(\langle \text{TC}, \text{density} \rangle \xrightarrow{\text{classify}} \text{MC} \right) \wedge \left(\langle \text{TC}, \text{MeanSpeed} \rangle \xrightarrow{\text{classify}} \text{TJ} \right) \wedge (v_{\text{range: MC} \rightarrow \text{TJ}} > k_{\text{range: TJ} \rightarrow \text{MC}}) \right) \\
\hline
\langle \text{TC}, \text{final} \rangle \xrightarrow{\text{classify}} \text{TJ}
\end{array}$$

Figure 20: Final traffic condition classified to TJ

4.5 Solution Validation and Experimental Results

4.5.1 Prototype software architecture

Figure 4 presented in chapter 3 - section 3.4.1 depicts the software architecture of our prototype, in which the matching model, traffic estimation model and traffic conditions classification are implemented based on our new approach.; The Data Consumer node performs the role of a user who is interested in the sensing activities and wishes to recognize the condition of a road; The Vehicular Sensing platform matches the consumer request whenever received with the appropriate set of collectors using cell Towers antennas while running the matching module. The platform then classifies the traffic condition into Free flow, Moderate Congestion or Traffic Jam through its traffic estimation module; The Data Collector node reflects the behavior of VanetMobiSim nodes and publishes its geographic data when requested only. Note that no voluntary publication of sensed data is sent to the sensing platform since the use of cell towers addresses such additional needs. Moreover, as the towers retrieve all kinds of sensors located in their coverage area, we randomly generate additional nodes to represent the sensors that are out of vehicles.

4.5.2 Testing Scenario

Figure 21: Full Scenario

Fig 21 illustrates our proposed sensing scenario. The data consumer first interacts with the platform by sending sensing request asking for traffic condition on specific road. The platform therefore searches for all towers that surround the requested road, sends them requests and waits for their replies. Once done, sensing request is sent to each sensor in the set gathered from the towers to get their current locations. The data collectors perform sensing operation to send the sensed data back to the platform. Since many collectors are not located on the desired road and some of them are not even in vehicles, the platform filters the set and sends another request only to the filtered nodes. When all responses are received, the platform runs both density and mean speed estimation algorithms to predict the real-time traffic status of the road and sends it to the consumer.

4.5.3 Experimental Results

The objectives of the conducted experiments are the following:

- Evaluate our proposed approach and indicate how accurate is the final inferred TC while using the vehicles selection model through mobile cell towers and the rule-based classification model for traffic condition.
- Compare the performance of the proposed approach with the one presented in chapter 3.
- Evaluate the overall system's performance included all the communications and processing overhead.

We used in our experiments the road topology of figure 17 generated from VanetMobisim and added four cell towers to cover the entire region. Each tower has 1 mile of diameter that covers the area shown in hexagonal shape. The simulated scenario is tested on road x of length 305 meters for the different traffic conditions where the real TC is observed visually from the demo showing the simulation traces.

4.5.3.1 Accuracy of Traffic Estimation

Figure 22 shows the performance of the traffic estimation module while considering the TC deduced from the density estimation only, the mean speed estimation only, and the TC inferred from the combined estimation based on the rule-based classification. Seven experiments are conducted for each classification of the Traffic Conditions.

In case of free flowing (FF), the density-based estimation shows 57.14% accuracy where 3 out of 7 experiments estimated the TC to MC. The 3 experiments inferred improper

Figure 22: Traffic estimation results

classification, yet their values were varying between 6 and 7 vehicles per 305 meters which are very close to the boundaries of FF category (0-5). As for the mean speed estimation and the final TC inferred, the 7 experiments were properly classified and hence achieving accuracy of 100%.

In case of Moderate Congestion (MC), the density based estimation shows 100% accuracy while the mean speed estimation shows only 28.57%. Among the 71% of the non-accurate experiments, some estimated a FF condition while others estimated a TJ. It was noticed that when the cars are moving all over the road, they could have somehow high speed and hence a FF condition is predicted. However, when the cars are gathered on a same segment of the road, the cars are stopped and a Traffic Jam is inferred. Regarding the TC inferred from the proposed combined rule-based estimation, all the conducted experiments estimated the real mean speed.

By analyzing the results in case of TJ classification, we can find that all the experiments show accurate results since the number of cars on the road is big and the speeds are low

along the entire road.

Our proposed approach in chapter 3 reflects the mean speed in km/h as a final estimation for the roads. In order to compare the results of our proposed approach, we classify the TC of the mean speed based on the thresholds of table 7. The resulting classifications are shown in Figures 23 and 24 along with the ones inferred from the proposed combined rule-based model.

Figure 23: Comparison of old and new approaches in FF traffic condition

Figure 24: Comparison of old and new approaches in MC traffic condition

Figure 23 illustrates the experiments reflecting FF condition, where 2 out of 7 of the TC inferred from the mean speed were improperly classified while the new approach predicts the correct classifications along all the experiments.

Figure 24 illustrates the experiments reflecting MC condition, where 3 out of 7 of the TC inferred from the mean speed were improperly classified while the new approach classifies the 7 experiments as MC.

We can show that the proposed combined rule-based classification model is capable of correcting the estimation towards the correct results in all conducted experiments.

4.5.3.3 Performance Analysis

We also conducted some load tests in terms of response time and network load to evaluate the overall system performance and the behavior of the implemented components. The response time is considered to be the elapsed time between the sending point of a request from a data consumer to the vehicular platform (message 1 in fig 21) and the receiving point of its response (message 15 in fig 21). The full response time encompasses the 10 seconds waiting time on the server side to send second requests to the targeted collectors for the sake of accomplishing the traffic estimation algorithm. The computation and communication response time does not consider the waiting time and is limited to the algorithmic computation of the different platform components and the communication time with data collectors. As for the network load, it is the size of the packets exchanged between the vehicular network platform and the data consumers and collectors.

Figure 25 shows the obtained load testing results which confirm the effectiveness of our approach. Both the system response time and network load are measured when the number of requests simultaneously sent from the consumers varies from 1 to 20 requests. In our

Figure 25: Load testing results for the full approach

simulation area, the microcell towers that are adopted could support up to 200 concurrent active users. We suppose that the 200 users are subscribed to our platform, and all of them participate in the sensing activities and publish their sensed data whenever requested. This will provide the best-case scenario in terms of the number of data publication messages and hence an accurate traffic estimation is predicted. However, it is the worst-case scenario in terms of overhead and delay in the response times. As shown in fig 25, the full response time ranges from 11.52 seconds for 1 sensing request to 54.68 seconds for a 20 sensing requests. As for the computation and communication response time, the values are obviously reduced by 10 seconds.

When receiving one request only, we first considered the case where the targeted road is covered by 1 cell tower (roads y in fig 17) and then a road covered by 2 towers (road x in fig 17). Therefore, the number of collectors is greater by 200 and 1.78 more seconds are required to send the final traffic decision to the consumer. As for the other requests, the targeted road is fully located in the coverage area of only one tower. It worth to mention

that using computers with better performance setup and supporting enhanced threading software can reduce significantly the response time and remain it constant when the number of sensing requests increases since the communications with the collectors are done in parallel.

Regarding the generated network load, the size of the exchanged packets ranges from 572.4 KB for 1 sensing request to 6141.7 KB for 2000 sensing requests. The network load's growth pattern can be explained by the fact that the more sensing requests are received, the more data collectors are targeted, which multiplies the number of messages exchanged through the system.

As noted before, each sensing request targets different tower in the studied area and thus the sensed data is gathered from various set of collectors, which reaches 4000 users for the 20 requests. If any new request comes for the same road, then the same inferred traffic condition is sent to the consumer without any delay. Moreover, if the new request targets new road that is covered by one of the towers already selected, then the first location for the collectors on the desired road is promptly deduced from their first response previously sent to the platform. Only the second request to the set of collectors is required and therefore the traffic decision is generated in few milliseconds.

4.6 Conclusion

In this chapter, we presented a new approach that extends our approach presented in chapter 3 in order to fully support on-demand strategy and enhance our sensing approach by estimating more accurate traffic condition. More precisely, we elaborated the matching

model by using cellular towers which helps identifying the nodes located in any area of interest as well as the traffic estimation model by using rule-based combined classification approach based on density and mean speed. Finally, we discussed the system validation and experimental results.

Chapter 5

Conclusion

The concept of sensing as a service implies the ability to offer sensory data to consumers, on demand, following a data utility-based model. In this thesis, we applied this concept to the area of intelligent transportation and have proposed a novel infrastructure-less vehicular sensing framework enabling the on-demand sensing of traffic conditions, about any area of interest, by relying on a selected set of mobile phone owners acting as data collectors. A multi-criteria participants' selection model and a traffic estimation model were proposed to support the operation of the vehicular sensing platform. The framework that was first introduced to achieve partial on-demand approach and estimate only the mean speed of the roads, has been further extended to support the fully approach while elaborating inference rules and classifying the traffic conditions to Free Flowing, Moderate Congestion, and Traffic Jam. The framework architecture was implemented using a combination of prototyping and traffic simulation traces generated using VanetMobiSim. As for the communication aspect, RESTfull web service interfaces were defined to enable the communication between the sensing platform and the users.

The obtained experimental results for the partial on-demand approach, show good mean speed estimation accuracy and resource efficiency, when compared to the traditional opportunistic continuous sensing approach. Among the lessons learned from this work, we note the following: The partial on-demand sensing is able to calculate mean speed values close to the ground truth in all test cases. There exists a tradeoff between the data collection frequency in vehicular sensing systems, and the accuracy of the traffic estimation results. The more frequently the data is collected, the more data points are acquired about the area of interest, and the more accurate are the mean speed estimation results. However, this high accuracy is associated with a cost to pay in terms of degraded system performance. Indeed, we observed that higher traffic estimation accuracy is associated with less system performance in terms of increased network load and response time. Furthermore, there exists an interesting trade-off between the complexity of the participants' selection approach used, the participation percentage, and the accuracy of the traffic estimation results. We observed that more complex participants' selection approaches that rely on contextual information to select the best participants (offering the highest quality sensing data records) can yield a high traffic estimation accuracy, even with a low percentage of vehicles participating in the sensing activity. On the other hand, if this contextual information is not available and simpler participants' selection approaches must be used, then a higher percentage of participants' vehicles must be employed to compensate for the lower quality in the sensed data records, if a high traffic estimation accuracy is to be maintained. Those results demonstrate that the on-demand participatory sensing approach can achieve high traffic estimation accuracy, while maintaining a good system's performance (in terms of reduced response time and network load).

Moreover, the experimental results for the extended approach explore the benefits that can be offered by the full on-demand participatory sensing approach in terms of achieving high traffic estimation accuracy and more resource efficiency, when compared to both the traditional continuous and the proposed partial on-demand approaches. It was remarkable that the approach is able to successfully infer the traffic status for all the conducted experiments when the traffic condition varies between the three categories: Free Flow, Moderate Congestion and Traffic Jam. The elaborated inference rules for the final traffic estimation were able to resolve any conflict in the categories predicted by the mean speed and density. Furthermore, the results show that the network load is low during the sensing processes as the messages between the platform and the users are only exchanged when responding to the consumers requests.

Finally, the following is the list of publications derived from the thesis work:

Conference Paper

- Wael AlRahal AlOrabi, Sawsan Abdul Rahman, May El Barachi, and Azzam Mourad.
"Towards On Demand Road Condition Monitoring Using Mobile Phone Sensing as a Service" *In the Proceedings in the 7th International Conference on Ambient Systems, Networks and Technologies (ANT)*, Madrid, Spain, May 2016

Submitted Journal Paper

- Sawsan Abdul Rahman, Azzam Mourad, May El Barachi, and Wael AlRahal AlOrabi.
"Using Participatory Mobile Phone Sensing for On-Demand Traffic Condition Monitoring". *IEEE Transactions on Vehicular Technology*

Draft Paper

- "Traffic Condition Estimation based on On-Demand Sensing with the support of Mobile Infrastructure".

Bibliography

- [1] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. T. Campbell, “A survey of mobile phone sensing,” *IEEE Communications magazine*, vol. 48, no. 9, 2010.
- [2] G. Dimitrakopoulos and P. Demestichas, “Intelligent transportation systems,” *IEEE Vehicular Technology Magazine*, vol. 5, no. 1, pp. 77–84, 2010.
- [3] B. Tian, Q. Yao, Y. Gu, K. Wang, and Y. Li, “Video processing techniques for traffic flow monitoring: A survey,” in *Proceedings of the 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2011. IEEE, 2011, pp. 1103–1108.
- [4] R. K. Ganti, F. Ye, and H. Lei, “Mobile crowdsensing: current state and future challenges,” *IEEE Communications Magazine*, vol. 49, no. 11, 2011.
- [5] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, and S. Madden, “Cartel: a distributed mobile sensor computing system,” in *Proceedings of the 4th international conference on Embedded networked sensor systems*. ACM, 2006, pp. 125–138.

- [6] P. Mohan, V. N. Padmanabhan, and R. Ramjee, “Nericell: rich monitoring of road and traffic conditions using mobile smartphones,” in *Proceedings of the 6th ACM conference on Embedded network sensor systems*. ACM, 2008, pp. 323–336.
- [7] N. Lu, N. Cheng, N. Zhang, X. Shen, and J. W. Mark, “Connected vehicles: Solutions and challenges,” *IEEE internet of things journal*, vol. 1, no. 4, pp. 289–299, 2014.
- [8] <https://feeneywireless.com/152-million-connected-vehicles-by-2020/>.
- [9] <https://www.emarketer.com/Article/2-Billion-Consumers-Worldwide-Smartphones-by-2016/1011694>.
- [10] R. Mizouni and M. El Barachi, “Mobile phone sensing as a service: Business model and use cases,” in *Proceedings of the 7th International Conference on Next Generation Mobile Apps, Services and Technologies (NGMAST), 2013*. IEEE, 2013, pp. 116–121.
- [11] B. Kerner, “The physics of traffic springer,” *New York*, 2004.
- [12] S. Abdelwahab, B. Hamdaoui, M. Guizani, and T. Znati, “Cloud of things for sensing as a service: sensing resource discovery and virtualization,” in *Proceedings of the Global Communications Conference (GLOBECOM), 2015 IEEE*. IEEE, 2015, pp. 1–7.
- [13] Y. X. Wazir Zada Khan, M. Y. Aalsalem, and Q. Arshad, “Mobile phone sensing systems: A survey,” *IEEE Communications Surveys Tutorials*, vol. 15, no. 1, pp. 402–427, 2013.

- [14] D. G. Adam and I. Oikonomidis, “Tarifa: Traffic and abnormalities road instructor for anyone,” in *Proceedings of the 2nd Student Workshop on Wireless Sensor Networks*, 2009.
- [15] R. Herring, A. Hofleitner, D. Work, O.-P. Tossavainen, and A. M. Bayen, “Mobile millennium-participatory traffic estimation using mobile phones,” in *CPS Forum, Cyber-Physical Systems Week 2009*, 2009.
- [16] A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H. Balakrishnan, S. Toledo, and J. Eriksson, “Vtrack: accurate, energy-aware road traffic delay estimation using mobile phones,” in *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*. ACM, 2009, pp. 85–98.
- [17] J. C. Herrera, D. B. Work, R. Herring, X. J. Ban, Q. Jacobson, and A. M. Bayen, “Evaluation of traffic data obtained via gps-enabled mobile phones: The mobile century field experiment,” *Transportation Research Part C: Emerging Technologies*, vol. 18, no. 4, pp. 568–583, 2010.
- [18] Z. He, B. Cao, and Y. Liu, “Accurate real-time traffic speed estimation using infrastructure-free vehicular networks,” *International Journal of Distributed Sensor Networks*, 2015.
- [19] X. J. Ban and M. Gruteser, “Towards fine-grained urban traffic knowledge extraction using mobile sensing,” in *Proceedings of the ACM SIGKDD International Workshop on Urban Computing*. ACM, 2012, pp. 111–117.

- [20] Y. X. Wazir Zada Khan, M. Y. Aalsalem, and Q. Arshad, "Mobile phone sensing systems: A survey," *IEEE Communications Surveys Tutorials*, vol. 15, no. 1, pp. 402–427, 2013.
- [21] T. Das, P. Mohan, V. N. Padmanabhan, R. Ramjee, and A. Sharma, "Prism: platform for remote sensing using smartphones," in *Proceedings of the 8th international conference on Mobile systems, applications, and services*. ACM, 2010, pp. 63–76.
- [22] B. Płaczek, "Selective data collection in vehicular networks for traffic control applications," *Transportation Research Part C: Emerging Technologies*, vol. 23, pp. 14–28, 2012.
- [23] D. Evans, "The internet of things: How the next evolution of the internet is changing everything," *CISCO white paper*, vol. 1, no. 2011, pp. 1–11, 2011.
- [24] "Jersey restful web service," <http://jersey.java.net/>.
- [25] F. Hao, M. Jiao, G. Min, and L. T. Yang, "A trajectory-based recruitment strategy of social sensors for participatory sensing," *IEEE Communications Magazine*, vol. 52, no. 12, pp. 41–47, 2014.
- [26] H. Li, T. Li, and Y. Wang, "Dynamic participant recruitment of mobile crowd sensing for heterogeneous sensing tasks," in *Proceedings of the 12th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), 2015 IEEE*. IEEE, 2015, pp. 136–144.

- [27] Z. Song, E. Ngai, J. Ma, X. Gong, Y. Liu, and W. Wang, "Incentive mechanism for participatory sensing under budget constraints," in *Proceedings of the Wireless Communications and Networking Conference (WCNC), 2014.* IEEE, 2014, pp. 3361–3366.
- [28] Z. Song, C. H. Liu, J. Wu, J. Ma, and W. Wang, "Qoi-aware multitask-oriented dynamic participant selection with budget constraints," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 9, pp. 4618–4632, 2014.
- [29] Z. Song, B. Zhang, C. H. Liu, A. V. Vasilakos, J. Ma, and W. Wang, "Qoi-aware energy-efficient participant selection," in *Proceedings of the 11th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON), 2014.* IEEE, 2014, pp. 248–256.
- [30] X. Li, W. Shu, M. Li, H.-Y. Huang, P.-E. Luo, and M.-Y. Wu, "Performance evaluation of vehicle-based mobile sensor networks for traffic monitoring," *IEEE transactions on vehicular technology*, vol. 58, no. 4, pp. 1647–1653, 2009.
- [31] "Jersey restful web service," <http://vanet.eurecom.fr/>.
- [32] "Jersey restful web service," <http://jax-rs-spec.java.net/>.
- [33] "Jersey restful web service," <http://www.postgresql.org/>.
- [34] Y. Lan and A. Harada, "Interference analysis and performance evaluation on the co-existence of macro and micro/pico cells in lte networks," in *Proceedings of the 75th Vehicular Technology Conference (VTC Spring), 2012 IEEE.* IEEE, 2012, pp. 1–5.

- [35] D. Ni, “Determining traffic-flow characteristics by definition for application in its,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 2, pp. 181–187, 2007.
- [36] D. H. Widyanoro and M. E. Munajat, “Fuzzy traffic congestion model based on speed and density of vehicle,” in *Proceedings of the International Conference of Advanced Informatics: Concept, Theory and Application (ICAICTA), 2014*. IEEE, 2014, pp. 321–325.
- [37] R. Sen, A. Cross, A. Vashistha, V. N. Padmanabhan, E. Cutrell, and W. Thies, “Accurate speed and density measurement for road traffic in india,” in *Proceedings of the 3rd ACM Symposium on Computing for Development*. ACM, 2013, p. 14.
- [38] V. L. Knoop and W. Daamen, “Automatic fitting procedure for the fundamental diagram,” *Transportmetrica B: Transport Dynamics*, vol. 5, no. 2, pp. 133–148, 2017.
- [39] W. Pattara-Atikom, P. Pongpaibool, and S. Thajchayapong, “Estimating road traffic congestion using vehicle velocity,” in *Proceedings of the 6th International Conference on Telecommunications ITS, 2006*. IEEE, 2006, pp. 1001–1004.
- [40] K. Slonneger and B. L. Kurtz, *Formal syntax and semantics of programming languages*. Addison-Wesley Reading, 1995, vol. 340.