RoadAhead - enhanced vision for traffic management

Miguel José Fitas Baúto

Thesis to obtain the Master of Science Degree in

Information Systems and Computer Engineering

Supervisor: Prof. Dr. Paulo Jorge Pires Ferreira

Examination Committee

Chairperson:	Prof. Dr. Daniel Jorge Viegas Gonçalves
Supervisor:	Prof. Dr. Paulo Jorge Pires Ferreira
Member of the Committee:	Prof.ª Dr.ª Teresa Maria Sá Ferreira Vazão Vasques

May 2016

Dedicated to my family.

Acknowledgments

First, I would like to thank my parents, Maria Adelina Fitas and Manuel Baúto, who gave me the strength to finish my master thesis, and to my brother João Baúto who help me through this process.

I also want to thank my thesis advisor, Prof. Doutor Paulo Jorge Pires Ferreira for his support, orientation and advices which contributed to the overall quality of this thesis.

Finally, I want to thank my colleagues André Ribeiro and Hugo Pimentel for their help and friendship along these years which I will never forget.

Resumo

Ao longo dos últimos anos o congestionamento rodoviário tornou-se num problema sério afetando todos os dias milhões de automobilistas. De forma, a atenuar este problema diversos sistemas têm vindo a ser desenvolvidos. A maioria destes sistemas utiliza geralmente uma arquitetura centralizada denominada Vehicle-to-Infrastructure (V2I). No entanto, esta arquitetura apresenta diversas desvantagens em relação ao seu tempo de resposta e tolerância a faltas. Assim sendo, a fim de resolver estes problemas novas arquiteturas tais como a Vehicle-to-vehicle (V2V) e Vehicle-to-vehicle-to-infrastructure (V2V2I) têm vindo a ganhar interesse nestes últimos anos. Contudo, estas arquitetura requerem a instalação de equipamento extra nos veiculos de forma a operarem corretamente.

Devido a estes problemas, propomos neste trabalho um sistema descentralizado denominado por RoadAhead que é por si só uma extensão e melhoria da arquitetura V2I, em relação ao tempo de resposta. Este sistema oferece uma visão mais ampla da condições de tráfego em tempo real, a fim de ajudar os automobilistas a tomar decisões mais prontamente quando comparado com a situação atual. Isto é realizado através da instalação de sensores rodoviários ao longo das estradas, que recolhem informações relevantes sobre o tráfego atual (e.g. a velocidade instantânea do veiculo). Posteriormente, estas informações são analisadas e apresentadas aos automobilistas, utilizando diodos emissores de luz dispostos ao longo das vias, que mudam a sua cor de acordo com as condições de tráfego (i.e. Verde, Amarelo, Vermelho).

Palavras-chave: sistemas de transporte inteligentes, congestionamento rodoviário, visibilidade aumentada, informação em tempo real, sensores rodoviários, diodo emissor de luz

Abstract

Over the past years traffic congestion has become a serious problem, contributing in a way that affects the lives of millions of vehicle drivers everyday. In order to mitigate this problem several systems have been developed. Most of these systems commonly use a centralized architecture denominated Vehicle-to-Infrastructure (V2I). However, this architecture has major drawbacks regarding response-time and fault-tolerance. In order to solve these problems architectures such as Vehicle-to-Vehicle (V2V) and Vehicle-to-Vehicle-to-Infrastructure (V2V2I) have gained interest in recent years. Nevertheless, these architectures require extra equipment attached-to or built-in vehicles to operate properly.

Due to these issues we propose in this work a decentralized system called RoadAhead which is by itself an improvement of the V2I architecture w.r.t response-time. This system offers a broader vision of the traffic conditions in real-time in order to help vehicle drivers make better decisions much in advance when compared to today's situation. This is done through the deployment of road sensors along the roads that collect relevant information about the current traffic (e.g. Vehicle instantaneous velocity). Moreover, this information is analyzed and displayed to vehicle drivers by using the capabilities of light emitting diodes (LEDs), which are disposed along the roads and change their color according to the traffic condition (i.e. Green, Yellow, Red).

Keywords: intelligent transport system, traffic congestion, increased visibility, real-time information, road sensors, light emitting diodes (LEDs)

Contents

List of Tables

List of Figures

Nomenclature

AVI Automatic Vehicle Identifiers

CMS Changeable Message Signs

COMPASS Freeway Traffic Management System

DCT Discrete Cosine Transform

DMS Dynamic Message Signs

EWMA Exponential Weighted Moving Average

GM-HMM Gaussian Mixture Hidden Markov Models

GPS Global Positioning System

HAR Highway Advisory Radio

HC Heavy Congestion

HMM Hidden Markov Models

HTML HyperText Markup Language

HTTP Hypertext Transfer Protocol

ITCS Integrated Traffic Control System

ITS Intelligent Transport System

JAX-WS Java API for XML-based webservices

LED Light Emitting Diode

LOC Level of Congestion

MC Mild Congestion

ML Maximum Likeluhood

MSVM Multi-class Support Vector Machine

OF Open Flow

OTM Ontario Ministry of Transportation

RFD Radio Frequency Signals

SMS Short Message Services

SOTIS Self-organizing inter-vehicle Traffic Information System

TCP Transportation Control Protocol

TIW Traffic Information Website

V2I Vehicle-to-infrastructure

V2V2I Vehicle-to-Vehicle-to-Infrastructure

VANET Vehicular Ad-hoc Networks

VICS Vehicle Information and Communication System

VIP Video Image Processor

VMS Variable Message Signs

WSDL Webservice Description Language

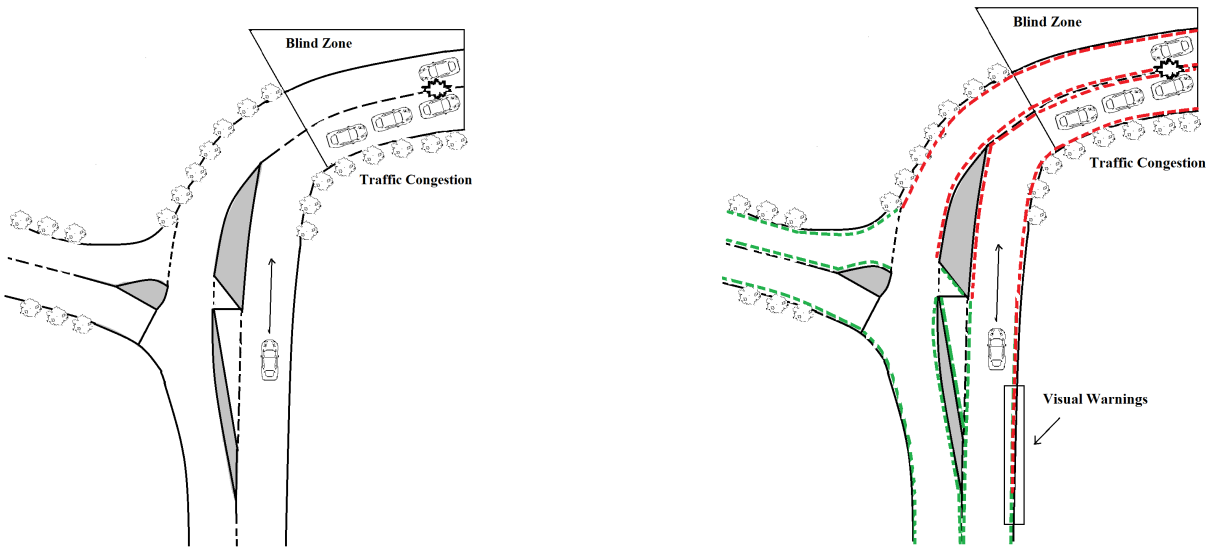
Chapter 1

Introduction

In modern society, vehicles are a part of people's life. As the number of vehicles increases, more the traffic situation becomes a serious and complicated issue that can lead to several problems such as traffic congestion (i.e. traffic demand exceeding the roadway capacity) [1].

To prevent these problems, transport infrastructures can be built but they are not feasible mainly due to cost issues [2]. In addition to this physical capacity constraint, external events can also have a major effect on traffic congestion. These include traffic incidents such as vehicle accidents and breakdowns, work zones, adverse weather conditions, and special events [3]. Moreover, traffic congestion and road incidents contribute negatively to the quality of life (e.g. waste of time for vehicle drivers, air pollution and fuel consumption), particularly when are vehicle drivers the cause of them.

The fact that vehicle drivers only view a portion of the road ahead is a major limitation while driving, mainly when adverse weather conditions (i.e. rain, fog, ice, snow, and dust), other vehicles ahead or intersections between roads obscure their vision.



(a) Vehicle driver reaching blind zone

(b) Vehicle driver with a broader vision

Figure 1.1: Different Scenarios.

Consider for example a vehicle driver who is traveling to a given location and reaches a road curve

with low visibility that prevents him from seeing what lies behind it.

As Fig. 1.1 (a) and (b) show, when approaching the road curve the vehicle driver is faced with heavy traffic congestion caused by an accident involving several vehicles. Since the vehicle driver only reacts to this situation while around the road curve, his reaction time will be longer which may lead yet to another accident. Ideally, such reaction time would be much shorter, if a warning were displayed to an electronic road panel or to the driver's car dashboard alerting him to the possibility of danger ahead.

1.1 Objectives

The goal of this dissertation is to develop a system called **RoadAhead** which offers a broader vision of the traffic conditions in real-time. This is performed by detecting existing vehicles on the road and displaying crucial information to vehicle drivers in an intuitive and visual manner as illustrated in Fig. 1.1 (b).

By observing the visual warnings (i.e. Green, Yellow, Red) displaced at the roadside, vehicle drivers are able to make better decisions about the route, speed, etc. much in advance (e.g. when compared to the current situation), preventing this way potential incidents that may occur.

The system is required to be reliable, efficient and accurate regarding the collection, management and dissemination of information to vehicles on the road. Other requirements of the system are to behave as promptly as possible i.e. fast response-time and to be non-intrusive to drivers.

In addition, it must be scalable and continue to operate properly independently of traffic density and penetration level (i.e. vehicles that participate on the system). Additionally, the system is required to preserve the vehicle driver's privacy by not collecting any information that could infer his exact position or any previously routes made by him.

Finally, it's necessary to ensure that the system operates correctly without any extra equipment attached-to or built-in vehicles in order to reach all possible drivers of a given road.

To achieve the goal and fulfill the above mentioned requirements, RoadAhead must deal with the following challenges: i) efficient gathering of information about vehicles that travel in a given road, ii) efficient management and treatment of the collected information, iii) dissemination of processed information, iv) fast-response time (soft real-time), and v) guaranty vehicle driver's privacy.

1.2 Existing solutions

Nowadays, there are systems that try to provide traffic related information to vehicle drivers. We describe them briefly and discuss their strengths and weaknesses; later, in Chapter 2, we analyze them in more detail.

The most used systems are based on Vehicle-to-infrastructure (V2I) architecture and examples of these are: **Integrated Traffic Control System (ITCS)**, **Vehicle Information and Communication System (VICS)** and **Freeway Traffic Management System (COMPASS)**.

Regarding ITCS [4], it's a large-scale traffic management system that realizes optimal traffic management for cities and environment by providing traffic information via various media and controlling signals based on the result of traffic information analysis collected by detectors on roads. This system has several strengths such as: high accuracy, high scalability by deploying additional sensors along the roads to be monitored, no requirement of any extra equipment attached-to or built-in vehicles and it's non-intrusive to vehicle drivers. However, some weaknesses are present regarding fault-tolerance, system response-time and the need to install, manage and maintain a large number of roadside sensors which might become expensive.

Regarding VICS [4], it's a system implemented in Japan that promptly provides the latest necessary road traffic information to vehicle drivers via car navigation equipment. VICS presents some strengths such as: high accuracy and delivers information with high level of detail. However, it has some weaknesses regarding fault-tolerance, the requirement of extra equipment attached-to vehicles which limits its availability to vehicle drivers that buy their specific equipment and the use of external equipment which might compromise driver's privacy.

Concerning COMPASS [5], it's an advanced traffic management system run by the Ontario Ministry of Transportation (OTM) to regulate traffic flow on the 400-series highways. COMPASS has several strengths such as high accuracy, high scalability and no need for extra equipment in vehicles but the most notable one is the ability to use incident detection algorithms to infer changes in traffic conditions. However, it shares weaknesses with both ITCS and VICS such as fault-tolerance and the requirement to install, manage and maintain a large network of road sensors/traffic cameras.

In addition to the systems described above there are others that are based on another architecture which is Vehicle-to-vehicle (V2I). One example is Self-organizing Traffic Information System (SOTIS) [6].

SOTIS is a system based on self-organizing inter-vehicle communication that uses equipment installed in vehicles to autonomously perform traffic analysis and broadcast the results to other vehicles within its transmission range. SOTIS has several strengths such as: delivering traffic information on the surrounding area with low delay and a high level of detail, fault-tolerance, and high scalability. However, some weaknesses are present regarding the requirement of extra equipment attached-to vehicles (e.g. GPS) which might compromise driver's privacy.

1.3 Solution

RoadAhead uses a decentralized architecture, as depicted in Fig. 1.2, which is based on the Vehicle-to-Infrastructure (V2I) architecture and can be applied to one or several road sections with variable size, being more suitable for roads that present higher traffic flow.

Our system divides the road network into several different zones, each one composed by three main components with well defined tasks: road sensors, local server and light emitting diodes (LEDs). Regarding road sensors, they are deployed along the roads and are responsible for the collection of traffic information (e.g. Vehicle instantaneous velocity). This information is then forwarded to the correspondent local server where it's analyzed.

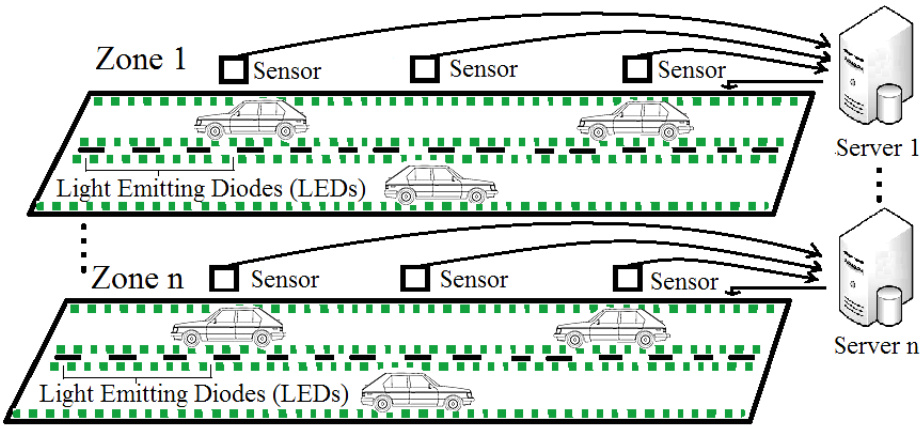


Figure 1.2: Decentralized V2I architecture

After this analysis, the local server is capable of inferring the current traffic situation, and delivers it through LEDs that are disposed parallel to the road. These LEDs show information to vehicle drivers by using a color grade system constituited by three different colors: Green, Yellow and Red, each one representing a traffic congestion state.

1.4 Thesis Structure

The remainder of this document is organized as follows. Chapter 2 describes the state of the art in the field of Intelligent Transport Systems. Chapter 3 presents our proposed architecture and how it fulfills the challenges that have been identified. In Chapter 4 we show how we implemented **RoadAhead**. Later, in Chapter 5 we present how we evaluate **RoadAhead** and discuss the overall results. To conclude, Chapter 6 summarizes work done and we discuss future work.

Chapter 2

State of the Art

This chapter addresses the related work suitable to the system mentioned in Chapter 1, focusing on the different aspects required to our solution.

Initially, several services that are supported by ITS (i.e. Intelligent Transport System) are identified. Then, architectures for ITS are described and analyzed. Furthermore, as depicted in Fig. 2.1, we describe three essential components of any ITS: detection mechanisms for vehicles on road, data management and treatment, and dissemination of the processed information.

Finally, we describe currently existing solutions stating briefly the advantages and disadvantages of each one.

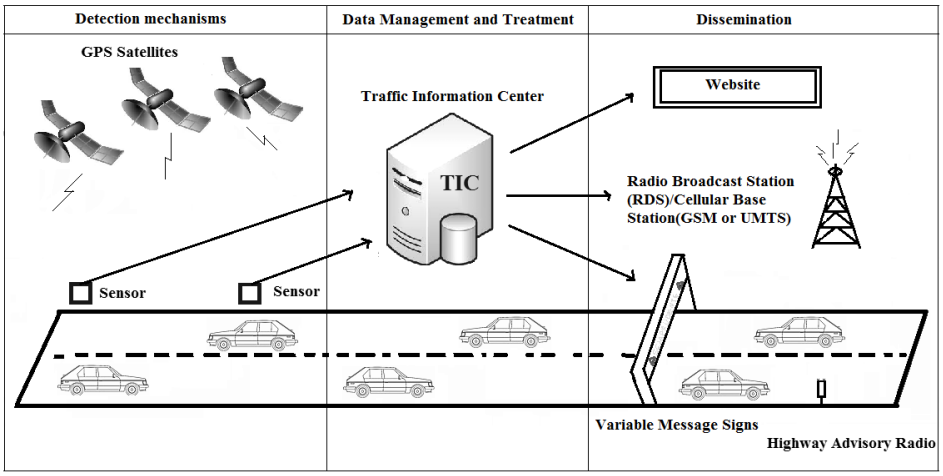


Figure 2.1: ITS Overview

2.1 ITS Services Characterization

ITS aim to support numerous services by applying information and communication technologies in vehicles or in roadside infrastructures to improve road safety, increase mobility, mitigate traffic congestion, and reduce fuel consumption and emissions.

In a report from the European Commission [7] and Kosch et al. [8] a wide variety of these services are described and organized within three groups: **Traffic Safety**, **Traffic Efficiency** and **Value-Added Services**.

Traffic Safety applications can be divided into two areas: Co-operative Awareness and Hazard Warning. The first one comprehends services such as lane changing assistance which gives information to vehicle drivers about vehicles in their proximity and the second one includes services such as advanced detection of slow vehicles ahead to prevent potential accidents or send post-crash warnings. Thus, vehicles involved in an accident send alert messages to other vehicles approaching the accident zone, helping in this way vehicle drivers to take proper precautions. Both these areas have a common key factor: that is how the information is collected, which instead of being just based on in vehicle sensing of local environment, it relies on vehicle-to-vehicle communications [9].

Concerning Traffic Efficiency, it focus on three distinct areas: Inter-Urban Traffic Efficiency, Urban Traffic Efficiency, and Freight and Fleet Applications. Inter-Urban Traffic efficiency considers the use of adaptive electronic traffic signs for incident detection and management, variable message sign display management, lane use management, speed management, and more. Additionally, there are other services that are part of Inter-Urban Traffic Efficiency such as road condition advice, route guidance and navigation. These services help vehicle drivers to avoid traffic congestions by informing vehicles of current traffic conditions, thereby providing in real time alternative routes that are less congested.

Regarding Urban Traffic Efficiency, it mainly covers traffic flow optimization by minimizing traffic delay and queues at intersections, giving preference to emergency vehicles and buses, and through speed management and incident detection. Furthermore, route guidance and navigation is also covered but in the context of urban environment. Finally, freight and fleet applications are those that cover e.g. the management of hazardous goods vehicles or the booking of delivery slots.

Finally, Value-Added Services provide services beyond those made freely available to drivers by accessing the local vehicle-to-vehicle or vehicle-to-roadside connections. Included in this area are services such as pre-trip and on trip journey planning, travel information, and location-based services.

2.2 Existing Architectures

2.2.1 Vehicle-to-infrastructure (V2I)

In Wischoff et al. [6], the authors state that conventional ITS employ an infrastructure-based approach as illustrated in Fig. 2.2. Several sensor-based traffic monitoring systems are installed along the road or on the roadside to collect information about the current state of the traffic (though, vehicles that act as mobile sensors, i.e. floating cars data [6], can also be used to observe the traffic conditions [10]).

Both these approaches transmit information periodically to a centralized Traffic Information Center, where all the received data packets are analyzed in order to achieve an overview of the current traffic situation. After this analysis, traffic reports are generated with information regarding the current traffic situation and packet into messages to the Traffic Message Channel [11]. These messages are forwarded

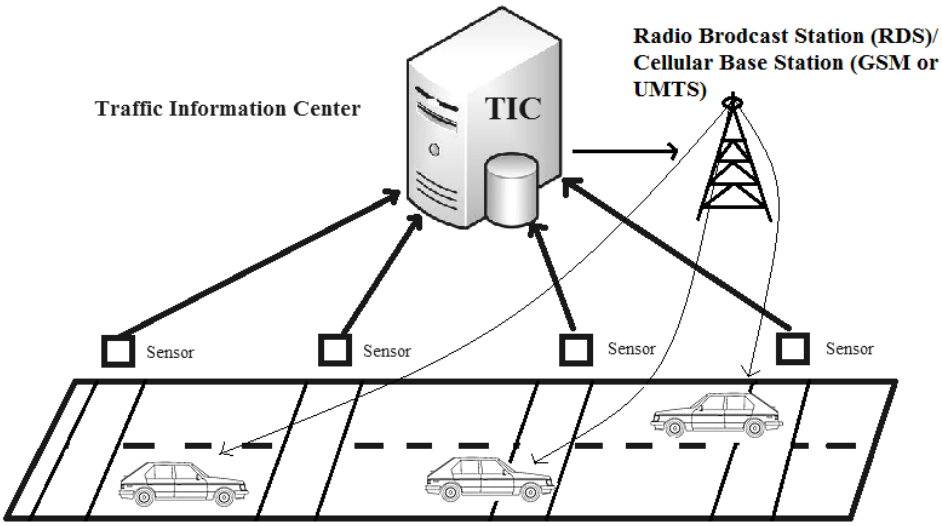


Figure 2.2: Vehicle-to-Infrastructure (V2I)

to the FM radio broadcast station that transmits them via radio data system to vehicle drivers. Alternatively, these messages can be delivered e.g. to cellular mobile phone networks, variable message signs or through websites.

A centralized architecture for delivery of information has interesting characteristics such as high accuracy **w.r.t** current state of the traffic situation, high scalability by deploying several sensors along roads where monitoring of the traffic is necessary, and reliability since it operates correctly regardless of traffic density. Moreover, this architecture usually doesn't require any extra-equipment attached to or built-in vehicles, respect driver's privacy, and can be considered in some cases non-intrusive to vehicle drivers due to specialized hardware that cause minimal disruption to the traffic. However, some limitations are present such as fault-tolerance since it uses a central unit that is a single point-of-failure. Additionally, this central unit is responsible for a relatively large area with limited bandwidth which prevents vehicle drivers from receiving specific traffic reports of the current location, receiving however general information. Finally, the distribution of traffic reports has high delay (e.g. commonly range 20-50 minutes) which is a major limitation **w.r.t** ITS due to the inability to provide real time information. Finally, regarding cost issues, this approach can be costly because it is necessary to install, manage and maintain a large number of roadside sensors.

2.2.2 Vehicle-to-vehicle (V2V)

Another approach for ITS is to use a decentralized architecture as presented in Fig. 2.3 where it is not required the installation of sensors along the road, roadside communication hardware nor a Traffic Information Center which costs less when compared to the V2I architecture [12]. In this case, vehicles act like mobile sensors, similarly to the FCD approach [13] in the centralized system and are equipped with communication devices (e.g. GPS [14], Wireless [9]) [10].

By using these devices, vehicles can communicate between each other by establishing Vehicular Ad-hoc Networks (VANET) [15]. Thus, they periodically exchange information about observed traffic

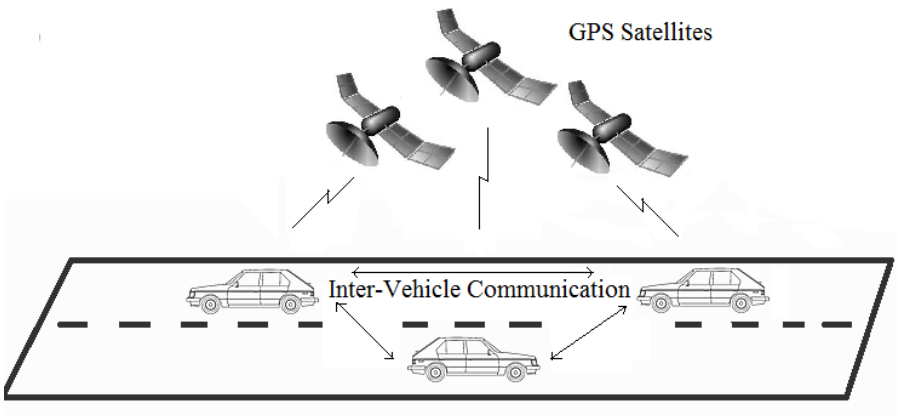


Figure 2.3: Vehicle-to-Vehicle (V2V)

conditions to other vehicles within their transmission range instead of transmitting that information to a centralized Traffic Information Center.After receiving information about the traffic situation from other vehicles, each vehicle performs an update of their own traffic information database deriving a traffic report for later exchange. Due to this, traffic reports are disseminated between vehicles which allow them to have an updated state of the current road situation.

Employing a vehicle-to-vehicle approach has several interesting characteristics such as fault-tolerance (i.e. the system continues to operate even if some vehicle fails), high scalability and a fast-response time. However, some drawbacks are present regarding traffic density and penetration level: if traffic density or penetration level is low, vehicles have difficulty in collecting updated information due to a significantly low amount of vehicles to exchange information within their range. Thus, if this is the case, traffic conditions can't be efficiently disseminated to large areas, which imply a high delay for retrieving distant traffic conditions and a low overall accuracy. Additionally, another limitation of this architecture regards driver's privacy due to the fact that each vehicle has attached to or built in equipment (e.g. GPS, Wireless) that can track in real-time vehicle's position and therefore infer routes previously made.

2.2.3 Vehicle-to-Vehicle-to-Infrastructure (V2V2I)

Miller [16] describes a V2V2I architecture as presented in Fig. 2.4, which is a hybrid of the V2V and V2I architectures (i.e. a combination of a Vehicular Ad-hoc Network (VANET) [15] with a Traffic Information Center).

In the V2V2I architecture, the transportation network is divided into several different areas. These areas are pre-configured which means that both the vehicles and the Traffic Information Center know where the vehicles are situated. Each area is assigned with a vehicle, known as Super-Vehicle that receives information (i.e. speed, location) from other vehicles in that area and communicates it to the Traffic Information Center, as well as, to other Super-Vehicles in adjacent areas. Thus, the Traffic Information Center can obtain a well-defined scenario of the traffic situation by analyzing all the information sent from all the Super-Vehicles within each area.

Regarding traffic density, the authors try to mitigate this problem by stating that each area size is

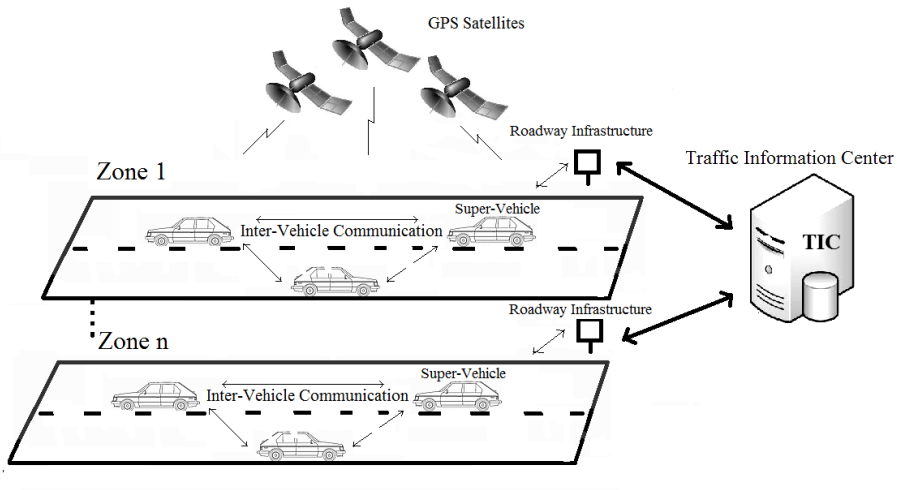


Figure 2.4: Vehicle-to-Vehicle-to-Infrastructure (V2V2I)

required to be small enough. This means that two vehicles at opposite points of the area must still communicate with each other. This is necessary due to the fact that one of these vehicles might be the Super-Vehicle, which means that it must receive all the information from all the vehicles within that area. In addition, this is also necessary to allow Super-Vehicles of adjacent areas to communicate with each other.

Therefore, the V2V2I architecture combines the characteristics of both the V2V and the V2I architectures, particularly the fault tolerant behavior of the V2V architecture and the high accuracy of the V2I architecture. Additionally, by using Vehicular Ad-hoc Networks (VANET), it reduces the bandwidth requirement for the roadway infrastructure including the Traffic Information Center.

However, V2V2I architecture has limitations in common with V2V architecture such as penetration level, vehicle driver's privacy and the requirement of extra equipment attached to or built-in vehicles. Finally, regarding cost issues this architecture requires a Traffic Information Center which implies a higher cost than V2V architecture but in contrast doesn't require a sensor-based network making it cost less than V2I architecture.

After all this said, Table 2.1 shows according to our system, a summary of the requirements that are satisfied by each of the previously described architectures. The most important thing that one can conclude is that none of these architectures fulfills all the requirements imposed by **RoadAhead**. However, one can see that V2I architecture best suits **RoadAhead** failing only to satisfy a fast response-time and least overall cost possible.

2.3 Detection mechanisms for vehicles on road

The efficient detection of vehicles is crucial to an ITS system that tracks in real time the road traffic conditions. Such a system must provide reliable information about the current state of the road to the vehicle drivers. To fulfill this requirement a proper detection of the vehicles presence, as well as, speed, volume and other relevant functions must be performed. Without this vital information the correct func-

Requirements	Architecture		
	V2I	V2V	V2V2I
Reliable	✓		✓
Accurate	✓		✓[a]
Fast Response-Time		✓	✓
Scalable	✓	✓	✓
Non-Intrusive	✓[b]		
Indifferent to Traffic Density	✓		✓
Indifferent to Penetration level	✓[b]		
Respects Driver's Privacy	✓[b]		
No Extra-Equipment in Vehicle	✓[b]		
Least Overall Cost Possible		✓	

Table 2.1: Fulfilled requirements per architecture

[a]with high penetration level
[b]depends upon the sensor-technology used for the detection of vehicles

tioning of the system is compromised, thus, the detection of vehicles is a major concern **w.r.t** accuracy and reliability of the system.

There are several mechanisms that allow the detection of vehicles and consequently of their traffic parameters. These mechanisms can be grouped into two distinct categories: **infrastructure-based detectors** and **vehicle-based detectors**.

2.3.1 Infrastructure-based detectors

Infrastructure-based detectors require installation of hardware by specialized personnel along several points on a given road where vehicles need to be detected and can be divided into two different groups: **intrusive detectors** and **non-intrusive detectors** [17, 18].

2.3.1.1 Intrusive detectors

Intrusive detectors are devices that are placed along the road usually in the pavement or underneath it causing temporary closure of roads for installation, operation and maintenance. Examples of such devices are [17, 19]: **inductive loops**, **magnetic sensors**, **piezoelectric sensors** and **pneumatic roadtube sensors**.

Inductive loops [17, 19] are electrically conducting wires that are isolated in the pavement. These wires are installed as a loop usually with a rectangular shape occupying most of the traffic lane. The loop is energized by an alternating current which is affected each time a vehicle enters in the electromagnetic field (i.e. inside the loop) [20]. The metallic parts of the vehicle are the reason for this disturbance because they absorb some of the energy, revealing then the presence of the vehicle. Through the years this technology evolved being one of the most used and can be very accurate when installed correctly.

Inductive loops are capable of inferring several traffic parameters (i.e. volume, speed, presence, occupancy, classification) due to a well-defined detection zone. The major limitations of such detectors are the pavement cut and temporary closure of roads for installation and maintenance, the several loops required when monitoring a given road (e.g. the estimation of vehicle's speed requires the passage

between two consecutive loops) and the cost issues that limit their scalability.

Magnetic sensors [19] are passive devices that detect the presence of metallic objects by sensing the variation (i.e. magnetic anomaly) in the Earth's magnetic field created by the passage of an object. Since almost all vehicles have metallic materials these devices can be used to detect them.

There are two types of magnetic sensors used to measure traffic parameters (i.e. volume, classification, presence, speed) [19]: two-axis fluxgate magnetometer and magnetic detector. The first one detects the presence of a vehicle by sensing the changes in the vertical and horizontal components of the Earth's magnetic field, while the second one measures the distortion in the magnetic lines of the flux due to a change in the Earth's magnetic field created by the vehicle.

These detectors share limitations such as: the temporary closure of roads for installation and maintenance, the need to cut the pavement and sometimes the deployment of several monitoring units to track the full lane. In addition, some magnetic sensors are unable to detect stationary vehicles and can be more expensive than induction loops which limits their scalability due to deployment costs.

Piezoelectric sensors [17] are devices that are able to measure the pressure signal after being pressed. These sensors have various forms but when used to detect vehicles they usually take the shape of a long narrow strip [21]. Whenever the vehicle's tires roll over the strip they deform it emitting a signal proportional to the pressure held, thereby detecting the presence of the vehicle.

Piezoelectric sensors gather more diversified information than others sensors since they collect it in a different way through the passage of the tire instead of the vehicle. Thus, they have the ability to distinguish vehicles with great accuracy (i.e. axle counting and spacing), gather improved speed, volume, occupancy and vehicle's weight.

In contrast, they share limitations from both inductive loops and magnetic sensors such as the temporary closure of roads for installation and maintenance, pavement cut, installation of multiple units along the road for proper monitoring and costs related issues (i.e. slightly higher than inductive loops) that jeopardize their scalability. Additionally, piezoelectric sensors are sensitive to temperature and to the vehicle's speed which can degrade their accuracy.

Pneumatic road tube sensors [17, 22] are devices that are constituted by rubber tubes and placed perpendicular to the traffic flow. Each time a vehicle's tires pass over the tube is propagated a burst of air pressure along it, closing then an air switch that produces a signal which is transmitted to a counter or to software for analysis.

Thus, these sensors are frequently used for collecting temporary information about the amount of vehicles, to classify them by axle counting and spacing and for planning and research studies. Further, some of these sensors have the ability to collect information for calculation purposes (i.e. vehicle gaps, delay stop intersections, stop sign delay, congestion flow rates).

The pneumatic road tube sensors have several interesting characteristics such as a quick installation for gathering of temporary or permanent information, low installation cost and simple maintenance.

However, they present some limitations regarding the inability to count accurately the number of vehicles when the roads have several lanes and when are constituted by curbs and/or gutters.

2.3.1.2 Non-intrusive detectors

Non-intrusive detectors are devices that are deployed along the roadside or above the roadway causing minimal disruption of the traffic for installation, operation and maintenance. Examples of such devices are [17, 19, 22]: **infrared sensors**, **microwave radar sensors**, **video image processor**, **ultrasonic sensors** and **passive acoustic sensors**.

Infrared sensors [17, 19, 22] are devices that are mounted above the roadway or on the roadside with a side-looking configuration. These sensors have the ability to convert the emitted or reflected energy into electrical signals that are analyzed to detect the presence of vehicles.

There are two types of infrared sensors: active infrared sensors and passive infrared sensors. Active infrared sensors illuminate the traffic lane (i.e. detection zone) with beams constituted by low power infrared energy that is reflected from vehicles and focused by an optical system onto an infrared-sensitive material mounted at the focal plane of the optics. Thus, they provide several traffic parameters such as vehicle presence at traffic signals, volume, speed, length, queue measurement and classification.

Regarding passive infrared sensors they don't emit energy of their own but instead they detect the energy that is emitted by vehicles, road surfaces, from atmosphere and other objects on their field of view. They are able to measure volume, occupancy and passage with a singledetection zone and when a multi-zone is deployed they can also measure speed and vehicle length.

Although, these sensors have limitations, active infrared sensors require temporary closure of road for installation, maintenance and lens cleaning. Regarding passive infrared sensors they have reduced vehicle sensitivity with adverse weather (i.e. rain, snow and fog). **W.r.t** cost issues active infrared sensors are more expensive than passive infrared sensors.

Microwave radar sensors [22] are devices mounted on the roadside that emit a signal in the form of beams. This signal is emitted to a specific area of the roadway from an overhead antenna in which its aperture is responsible for the beam width that defines the detection zone.

When a vehicle passes through the detection zone, the signal is reflected in the vehicle and received by the antenna. The time delay from sending and receiving the signal is used to calculate the distance to the detected vehicle [17]. Additionally, radar sensors can also infer several traffic parameters such as volume, speed, presence, classification, occupancy and length.

These sensors are capable of detecting stationary vehicles and can monitor multiple zones due to its range ability. However, they have some limitations regarding the place of installation (e.g. areas with barriers, fences and other obstacles) where may occur dead detection zones and "ghost" vehicles, which might affect their accuracy.

Video Image Processor (VIP) [19, 22] systems are typically constituted by one or more cameras, a microprocessor that processes the captured images and by software that interprets and converts the images into traffic data.

These systems are mounted on the roadside or on a centralized location of the roadway and detect the presence of vehicles by automatically analyzing successive frames of a given traffic scene in search of differences between them. Three types of VIP systems have been developed: tripline, closed-loop tracking and data association tracking.

Tripline systems allow the user to define a limited number of detection zones in the field of view of the camera. In this system, when a vehicle passes on one of these detection zones, it's identified by noting changes in the pixels of the roadway image in its absence compared to the pixels of the image with the vehicle in it.

Regarding closed-loop tracking systems, they are an extension of the tripline system and are capable of detecting vehicles along larger roadway sections. This system has the particularity of detecting vehicles by continuously tracking them through the field of view of the camera. Thus, several detections along the track must be made to validate the detection of the vehicle.

Concerning data association tracking systems, they identify and track a specific vehicle or group of vehicles as they appear in the field of view of the camera. This system detects the vehicles by searching in the image for unique areas of connected pixels that are tracked in each frame to produce afterwards data for the selected vehicle or vehicle groups.

VIP systems are able to collect several traffic parameters such as volume, speed, presence, occupancy, density, queue length, dwell time, headway, turning movements, acceleration, lane changes and classification. Additionally, these systems need to be installed and calibrated properly to perform well and to gather accurate data in poor lighting conditions. Furthermore, they are generally costeffective only if several detection zones exist within the field of view of the camera.

Ultrasonic Sensors [17, 19] are devices that are mounted with a side-looking configuration or with an overhead position on the roadway. They transmit pressure waves of sound energy at a frequency between 25 and 50 KHz which are inaudible to humans.

Generally, these sensors use pulse waveforms that measure the distance to the road surface and vehicle surface by detecting the portion of the transmitted energy that is reflected back to the sensor from a predefined area by the transmitter's beamwith. When the distance to the road surface is different from the measured without a vehicle, the sensor infers this as the presence of a vehicle.

Ultrasonic sensors are able to infer traffic parameters such as volume, presence and provide occupancy information. Regarding their installation, they don't require an invasive pavement procedure (i.e. pavement cut) and some models have the ability to monitor multiple lanes. However, they are susceptible to temperature changes and extreme air turbulence which may affect their performance. Additionally, large pulse repetition periods may degrade occupancy measurement due to vehicles traveling at moderate to high speeds.

Passive acoustic array sensors [17, 19] are devices that detect acoustic energy or audible sounds emitted by vehicles on road (e.g. friction of vehicle's tires on road, vehicle engine).

When a vehicle passes through the detection zone, an increase in the sound is detected by the signal-processing algorithm and the presence of the vehicle is identified. When the vehicle leaves the detection zone, the sound level returns to a state below the detection threshold and it's inferred that the vehicle has leaved. Also, it's important to refer that all sounds outside the detection zone are attenuated to not interfere with a proper detection.

These sensors have several interesting characteristics such as the capability of measuring vehicle passage, presence and speed, not requiring invasive pavement procedures for a proper installation and

in some models they have the ability to monitor multiple lanes.

Nevertheless, some limitations are present regarding their operation at cold temperatures that affect the accuracy of the collected data and the fact that some models aren't recommended for traffic that is constantly stopping or too slow.

2.3.2 Vehicle-based detectors

Vehicle-based detectors are those that can be attached to or built-in vehicles, not requiring invasive procedures in the road but causing in some cases minimal disruption of the traffic for their installation. Examples of such devices are: **Global Positioning System** and **Automatic Vehicle Identifiers**.

Global Positioning System (GPS) is a satellite based navigation maintained by the U.S Department Of Defense that is frequently used in road navigation providing fast, flexible, and relatively inexpensive data to determine vehicles position and velocity in real time.

The current constellation consists of 28 geostationary satellites with a period of 12 hours [23] and is based on loosely three dimensional positioning of satellites using trilateration related techniques. GPS employs two fundamental observables for positioning and navigation, the codephase or pseudo-ranges and carrier-phase. It can provide information about location data in terms of latitude, longitude, elevation and UTC time. Based on these spatial and temporal data one can determine traffic information such as travel time, travel speed, travel distance and delay [24].

GPS is less effective in urban areas because of the poorness of the satellite coverage due to tunnels, high buildings, electronic interferences, foliage, mountains, etc... [25] which can affect its accuracy. Moreover, GPS has one major limitation regarding the disclosure of trajectory data which may compromise the vehicle driver's privacy (even if trajectory data is broadcasted in an anonymous manner, it's still possible to infer individuals from the trajectory data [26]).

Automatic Vehicle Identifiers (AVI) is a system that combines different components such as AVI readers and AVI tags or transponders in vehicles. AVI readers such as antennas are located on the roadside, above the roadway or as part of an electronic toll collection booth.

These antennas transmit radio frequency signals (RFD) within a capture range across one or more traffic lanes. When a vehicle enters the detection zone (i.e. antenna's capture range), the transponders in the vehicles respond to the radio signal, and to its unique ID is assigned a timestamp by the AVI reader, thus detecting the presence of the vehicle [24].

AVI system has the capability to infer several traffic parameters such as volume, vehicles position, classification travel time and average vehicle speed between sequential antenna locations. This system is able to collect large amounts of data by using minimal human resources but has drawbacks in common with GPS regarding vehicle driver's privacy (each vehicle has a unique ID that can be traced between sequential antenna locations).

Table 2.2 presents an overview of the collection methods that have been previously described, specifying for each one the traffic parameters that can infer and their overall cost. Since our system is required to be non-intrusive and available to all vehicle drivers one can note that both intrusive and vehicle-based

detectors can be discarded and only non-intrusive detectors are able to fulfill the above mentioned requirements. Thus, by observing Table 2.2 one can conclude that the detector that best fits **RoadAhead** is the microwave radar sensor because when compared to the remaining non-intrusive detectors it's able to infer several essential traffic parameters with a lower overall cost, being so considered the most cost-effective.

Detector Type		Collection Method	Volume	Presence	Speed	Occupancy	Classification	Travel Times	Position	Overall Cost[a](each in 1999\2000 U.S. $)
Infrastructure-based	Intrusive detectors	Inductive Loops	✓	✓	✓[b]	✓	✓[c]			Low[d]($500-$800)
		Magnetometer	✓	✓	✓[b]	✓				Moderate[d]($900-$6,300)
		Magnetic Induction Coil	✓	✓[e]	✓[b]	✓				Low to Moderate[d]($385-$2,000)
		Piezoelectric	✓	✓	✓[b]	✓	✓			Low to Moderate($1,100-$4,000)
		Road Tubes	✓	✓	✓[b]	✓	✓			Moderate($1650-$6,500)
	Non-intrusive detectors	Active infrared	✓	✓	✓[f]	✓	✓			Moderate to High($6,500-$14,000)
		Passive infrared	✓	✓	✓[f]	✓				Low to Moderate($700-$1,200)
		Microwave Radar	✓	✓[g]	✓	✓[g]	✓[g]			Low to Moderate($700-$3,300)
		Video Image	✓	✓	✓	✓				Moderate to High($5000-$26,000)
		Ultrasonic	✓	✓		✓				Low to Moderate($600-$1,900)
		Passive Acoustic Arrays	✓	✓	✓	✓				Moderate($3,100-$8,100)
Vehicle-based		GPS			✓		✓	✓	✓	-
		AVI	✓		✓		✓[h]	✓	✓	-

Table 2.2: Traffic parameters inferred by several collection methods and their overall cost [19, 27]

[a] Installation, maintenance, and repair costs included.
[b] Speed can be measured by using two consecutive sensors with a known distance apart or estimated from one sensor, the effective detection zone and vehicle lengths.
[c] With specialized electronics unit containing embedded firmware that classifies vehicles.
[d] Includes underground sensor and local detector or receiver electronics.
[e] With special sensor layouts and signal processing software.
[f] With multidetection zone passive or active mode infrared sensors.
[g] With microwave radar sensors that transmit the proper waveform and have appropriate signal processing.
[h] The information must be on the tag.

2.4 Data Management and Treatment

The deployment of a large number of sensors along roads associated to the need to provide reliable information about the current state of the traffic conditions require the analysis of a huge amount of data produced by these devices. However, to efficiently use and manage these heterogeneous data, it's a big challenge, and even bigger for real-time use cases and applications [28].

Vanajakshi [24] divides the data analysis in data cleaning, fusion and analysis. A similar approach is presented in Lopes et al. [28] where the authors claim that the data collection task is an essential component of the data analysis process but it's not the only one. As Fig. 2.5 shows, together with data preprocessing and fusion, implements a crucial step to provide stable, coherent and efficient traffic data for upcoming application and systems.

To a better understanding of the data analysis process it's necessary to explain the purpose of data preprocessing and fusion. In this way, data preprocessing is used to mitigate problems that are present in real world raw sensor data. This data is highly susceptible to noise, missing values and inconsistent data due to malfunction of sensors, measurement errors, and data link errors or simply

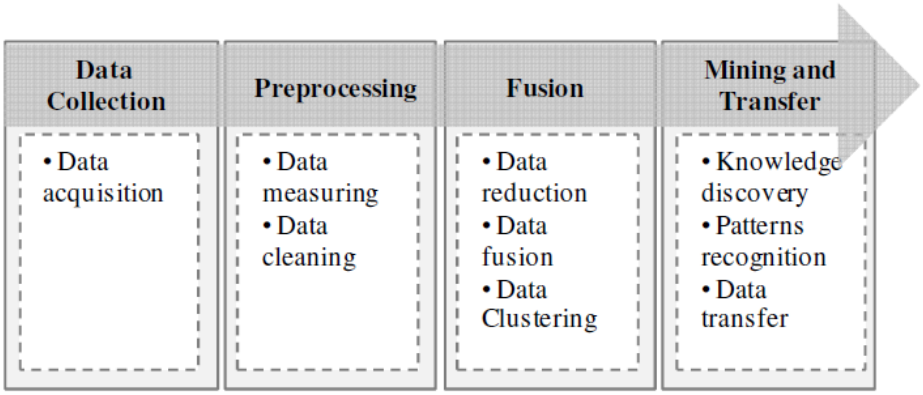


Figure 2.5: Data Analysis Process [28]

because of the huge data sample size. It's important to notice that low quality data will lead to poor data processing, whichever downstream traffic data application is to run. Furthermore, due to measurement errors inferred data can present values that are physically impossible such as negative vehicle counts or average speeds, unless wrong way driving incidents occur [28]. Further, as previously stated, huge amounts of data are produced by sensors deployed along roads. This inhibits the transmission of all this data to a Traffic Information Center to be analyzed. Thus, either the huge amounts of data are analyzed locally by deploying section stations or the amount of data is reduced by fusion which is the process of combining together the information collected by a variety of heterogeneous sensors into a single data set [29].

After all this said, one can conclude that the ability to combine all the data collected by sensors or other devices in an appropriate manner is a key issue for information systems solution providers and a challenge for the data processing systems and methods. Due to this, in Lopes et al. [28] the authors describe several methods to mitigate these problems in order to support real-time applications where data completeness, consistence, performance and reliability are required.

Furthermore, traffic and congestion estimation and forecasting methods can use this data, through the development of algorithms and techniques that combine all this unrelated data in order to be used by predicting systems.

2.5 Estimating Road Traffic Congestion

Currently, there is no standard recognized method to measure traffic congestion quantitatively on roads and at intersections [30]. However, the overwhelming majority adopts delay or the length of the vehicles delayed, which can't reflect the essential state of traffic congestion completely, and therefore can't provide accurate direction to improve the traffic state [31].

In order to mitigate this issue, many studies develop metrics for congestion measures. The two most cited reports are the Texas Transportation Institute's Urban Mobility [32] and the National Cooperative Highways Research Program Report 398 [33]. Examples of parameters used to indicate congestion include travel delay, travel time index, travel rate, travel rate index, and speed reduction index. One can

note that several metrics are required because the public and transportation engineer's generally do not agree on a single, widely accepted definition of congestion based on one and only measurement [34]. Additionally, a single measurement is usually not sufficient to express all crucial and incidental aspects of congestion.

Popular traffic report services such as traffic radio, real-time web report, and variable message signs (VMS) commonly use a single severity index to report congestion condition (e.g. web reports and VMS use the red (traffic jam) - yellow (slow moving) - green (free flow) colors to indicate three levels of traffic congestion while the radio/newscasters uses terms like "light traffic" and "heavy traffic") [35]. The congestion level is usually inferred by reporters looking at traffic cameras, traffic police at intersections, or drivers on roadways but since the inference is based on individual perception and feelings (i.e. in the same road condition, some may feel that the road is heavily congested, while others may feel that the road is only slightly congested), there is no consistent definition of the degree of congestion. Thus, in the past literature, several studies aim to tackle this problem by proposing methods that can be used along with computational efforts to determine levels of congestion from traffic flow information.

Porikli et al. [36] propose an unsupervised, low-latency traffic congestion estimation algorithm that operates on MPEG video data. They extract congestion features directly in the compressed domain, and employ Gaussian Mixture Hidden Markov Models (GM-HMM) to detect traffic condition. First, they construct a multi-dimensional feature vector (in order to capture the speed and density of vehicles) from the parsed Discrete Cosine Transform (DCT) coefficients and motion vectors. Then, they train a set of left-to-right HMM chains corresponding to five traffic patterns (stopped (S), heavy congestion (HC), mild congestion (MC), open flow (OF), and empty (E)), and use a Maximum Likelihood (ML) criterion to determine the correspondent traffic state from the outputs of the separate HMM chains. Their tests show that the proposed method is computationally efficient and modular, and that the feature vector is invariant to different illumination conditions (e.g. sunny, cloudy, and dark). Furthermore, experimental results show that the precision rate of the presented algorithm is very high (around 95%).

Krause et al. [29] use a Fuzzy Logic system designed with fuzzyTECH [37] to determine six continuous levels of congestion by using traffic volume, speed and estimated density as inputs into their fuzzy inference systems. Fuzzy Logic is a model that matches the relationship between inputs and outputs based on the probability theorem [38]. The Fuzzy Logic process has two main parts: 1) input and output membership functions, whose range is manually defined to fit with input/output logics, and 2) fuzzy rules which are manually designed by a programmer according to his/her expertise on solving particular problems. Fuzzy Logic can handle situations where there are uncertainties involved, such as problems that depend on human feeling and experience. Therefore, it's suitable for reporting road traffic where different people may feel differently in the same congestion situation. Krause et al. shows that the new method is faster and more reliable than conventional detection criteria, and that in comparison with existing methods, the fuzzy logic congestion detection guarantees shorter indication times. Moreover, Fuzzy logic systems generated by fuzzyTECH require low computational effort. In addition, the new method only uses existing detection equipment and compensates for the uncertainties of common traffic detection by fuzzy logic with the possibility of being installed on existing traffic control systems.

Qi-Liang [39] uses a Fuzzy Logic approach and divides the city road-network traffic congestion into road congestion and cross congestion, classifying it into five discrete states :jam, congestion, little congestion, little fluency, fluency. Then, it selects the appropriate evaluation indexes for traffic congestion (i.e. queue length, parking rate secondary, saturation, run delay, time occupancy, and speed) and builds the evaluate index system with analytic hierarchy process, and the corresponding integrative fuzzy evaluation model. Finally, the fuzzy evaluation method is used to evaluate the traffic states of a typical city roadnetwork.

Pongpaibool et al. [38] use fuzzy inference systems, but their study differs from the previous ones in that they examine both fuzzy logic and adaptive neurofuzzy systems [40] to evaluate levels of road traffic congestion. The motivation behind their interest in adaptive neuro-fuzzy is that the effectiveness of the manually tuned fuzzy systems depends highly on the fuzzy rules created by programmers. Therefore, these rules must be adjusted by trial-and-error methods until the results are satisfied. Thus, the adaptive neuro-fuzzy can solve this problem by automatically creating fuzzy rules according to given inputs and outputs. In order to correctly evaluate these two methods, they limit the traffic status to only three levels - free flow, slow moving, and heavily congested. In addition, they obtain real traffic information consisting of vehicle volume and average speed per minute which are fed into their fuzzy systems. Their experiments show that is possible to use fuzzy logic to evaluate current traffic condition and that the manually tuned fuzzy logic system achieves much higher accuracy than the adaptive neuro-fuzzy system (this is because the fuzzy logic system can capture human expertise better through manual adjustment of the membership functions).

Lu et al. [41] present a new method to evaluate congestion from traffic flow information based on fuzzy logic and define a new index named level of congestion (LOC) as a continuous variable from free flow to traffic jam. An adaptive neuro-fuzzy inference system is adopted as the training system to train the fuzzy logic rules in order to estimate LOC by inputting mean velocity and density. Finally, they analyze the system and show that is coherent with human perception.

Pattara-Atikom et al. [35] propose an algorithm that uses two quantifiable measurements - vehicle velocity which is the only measured data in order to minimize the amount of data required and congestion duration. With a simple threshold technique, vehicle velocities can be classified into three congestion levels indicating degrees of congestion (i.e. red, yellow, and green). To remove any precipitate congestion event indication (i.e. occurred when the velocity changes shortly and abruptly) the congestion duration is used to classify and eliminate a transient congestion event from a steady congestion event. The steady congestion event is the final output to be reported to the public. The algorithm was evaluated by two groups of independent people, one to train and another to validate the algorithm. Their experiments show that the algorithm has several interesting characteristics such as: estimating congestion with a error around a half of a congestion level, requiring minimal input data, the ability to report a consistent definition of congestion, and being quantifiable and applicable to automated report.

Deng et al. [42] propose a novel pattern-based approach to model the clustering and classification of traffic state. First, fuzzy-set clustering method (i.e. K-Means cluster method which was first brought out by James Macqueen [43]) is used to divide the traffic state into a number of patterns. Then, multiclass

support vector machine (MSVM) is applied to estimate these states with real-time traffic data. Their experiments show that proposed approach is promising for the dynamic estimation of road traffic state and can provide forecasted congestion information for the traffic control system and traffic guidance system.

Finally, Kaysi et al. [44] discuss the use of historical data profiles as part of the development of a congestion prediction algorithm for a real time driver guidance system in West Berlin. Their approach involves the creation of a historical data profile to make the prediction to the future time steps.

From all previously described methods and studies, there is only one that can't be applied to our system which is the one proposed by Porikli et al. [36]. This method isn't suitable because it operates only on video image data, thus being incompatible with other data collection methods (e.g. Microwave Radar Sensors).

With that said, we concluded that all the other methods using a fuzzy logic approach or a simple threshold technique are acceptable to be used by our system. Regarding fuzzy logic approach, we consider that it's appropriate because as said before, it can handle situations where there are uncertainties involved (e.g. human feelings) which means that it can be used to report road traffic with high accuracy where different people may feel differently in the same congestion situation. Finally, concerning the algorithm proposed by Pattara-Atikom et al. [35], we think that it's also a good choice because it requires minimal computational effort, as well as, minimal input data (i.e. requires only velocity measurement) which to our system is considered to be very important in order to provide an effective response-time with a correct traffic congestion answer.

2.6 Information Dissemination

The efficiency of an ITS system doesn't rely only on the collection and analysis of the traffic data, but also on the dissemination of the inferred information using this data to the intended receivers, which are those that are interested in it (e.g. vehicle drivers, ITS personnel, road users). Thus, data dissemination concerns the delivery of traffic information to these receivers, while meeting some requirements (e.g. low delay, high reliability, low message passing overhead).

Several systems are used for relaying traffic information to vehicle drivers such as **Variable Message Signs (VMS)**, **Highway Advisory Radio (HAR)**, **Traffic Information Websites**, **Short Messaging Services (SMS)**, **GPS Navigation Systems**, **Radio/Television Stations** and other modern media tools. Further, these systems are described stating briefly their characteristics, benefits and issues.

Variable Message Signs (VMS), also known as changeable message signs (CMS) or dynamic message signs (DMS), are devices that are mounted above the roadway or at the roadside, which convey traffic related information to vehicle drivers, in a visual manner, about events that might affect their travel experience and safety [45, 46]. In addition, VMS provide mandatory or advisory information to vehicle drivers and can be used for many different purposes with the potential benefits of reducing vehicle driver's stress, travel time and increasing traffic safety. However, all these benefits come with a high cost (i.e. $41K - $101K per unit, depending on the type of sign, and an operation and maintenance cost of

$2K - $5K) which is the main barrier to implement them, thus affecting their scalability.

Highway Advisory Radio (HAR) is a special radio tool that provides to traffic operating agencies the capability to communicate, through audio messages, traffic and travel related information to vehicle drivers by using the vehicle AM radio receiver. HAR messages are transmitted from low-power roadside transmitters (with a cost between $15K - $36K and an operation and maintenance cost of $0.6K - $1K). The vehicle drivers approaching an HAR site are advised of its existence by advanced highway signs (with a cost between $4K - $8K per unit) which tells them the correct tuning dial to receive the message. HAR messages are generally recorded for continuous repetition and its length is adjusted to allow vehicle drivers to receive the messages at least twice while passing through the station's coverage zone. Moreover, agencies operating HAR must monitor and maintain the system, as well as, change the message content as the roadway conditions change. HAR can provide to vehicle drivers warnings, advisories, and directions, or other non-commercial material of importance [47]. However, in comparison with visual systems, it has some drawbacks regarding the requirement to vehicle drivers tune their radios at a given frequency, the fact that HAR system relies heavily on how well designed the messages are which involves a complex and slow process mainly due to principles and guidelines that should be followed by qualified personnel, and finally, the fact that complex audio messages may be misunderstood by vehicle drivers which can lead to undesired situations (e.g. congestion, accidents, etc..).

Traffic Information Websites are web portals that provide traffic and travel related information about the current road condition to vehicle drivers. This information can be visualized on a map by selecting a given location or route of interest, where afterwards, all major events happening such as existing accidents, traffic jams, or construction, etc.. are displayed. Traffic Information Websites are generally used for pre-tip planning but can also be utilized for on trip planning by making use of existing technologies that can be transported in vehicles such as laptops or mobile phones with internet connection. Despite all the wide range of information, Traffic Information Websites have some crucial limitations when compared to other systems (e.g. Variable Message Signs) which can help in real-time vehicle drivers. These limitations are as follows: requirement of extra-equipment in vehicles in order to access the traffic and travel information, disruption of the service due to internet connection problems, and finally, may decrease the perception of the road due to the interaction with the equipment.

Short Messaging Service (SMS) is a text messaging service used by traffic operating agencies to inform vehicle drivers about the traffic conditions in real-time. Usually, to take advantage of this service, vehicle drivers are required to subscribe to it by sending a text message to the service provider. After the subscription confirmation, vehicle drivers are able to receive the latest traffic news by sending text messages with predefined tags which provide access to a wide range of information. This type of service offers an easy access to traffic information which can be used for pre-trip planning or on trip planning. However, it presents some limitations regarding the subscription or message charge, the requirement of extra-equipment (e.g. mobile phones), the need to write and send messages which is prone to human error, and the constant interaction with the equipment which may reduce the driver's attention.

GPS Navigation Systems are able to provide traffic services which offer vehicle drivers up-to-date information about traffic conditions. These services are available in an on growing number of metropoli-

tan areas and warn vehicle drivers about specific incidents, including accidents and constructions. The main limitations regarding this kind of system is the fact that it requires additional equipment attached-to or built-in vehicles (some even require e.g. mobile phones with wireless connections) and the disclosure of trajectory data which may compromise the vehicle driver's privacy (even if trajectory data is broadcasted in an anonymous manner, it's still possible to infer individuals from the trajectory data [26]).

Radio/Television Stations can also be used to report traffic information through public announcements made by announcers. These announcements provide general traffic information about the most busiest routes and are usually made on peak hours. This type of service has some drawbacks such as not providing traffic information in real-time, not allowing vehicle drivers to receive traffic information about a particular road and requiring extra-equipment (i.e. radio receiver and television).

2.7 Relevant Systems

In this section, we describe existing systems that have some similarities with the system that is proposed in this document. Although, some of the systems might not share the same goal as ours they provide important insights to the development of **RoadAHead**.

Integrated Traffic Control System (ITCS) [4] is a large-scale traffic management system implemented in Japan that achieves safe and smooth road traffic. The traffic control system has three main functions: collection, processing and presentation of traffic information. These basic functions are described as follows. Roadside traffic sensors collect the traffic information, as well as, helicopters, patrol cars and police motorcycles which supervise the road network and send their information to the Traffic Control Center. Additionally, police stations, police boxes and television cameras complete the collection of traffic information. Afterwards, all collected information is processed by the Traffic Control Center in the following order: control of the intervals of the signal (red/green) in ratio to traffic, displaying traffic jams on the Central Display Board, permanent update of the telephone and facsimile information service and the traffic news transmitter, displaying where traffic accidents happen and which streets are closed, and finally, exchange of traffic information with neighboring Traffic Control Centers. The processed information is then delivered to vehicle drivers via various media such as: variable message signs, traffic signals, traffic news transmitter, etc.

ITCS presents several strengths such as: high accuracy **w.r.t** current state of the traffic situation, high scalability by deploying additional sensors along the roads to be monitored, no requirement of any extra-equipment attached to or built-in vehicles and it's non-intrusive to vehicle drivers. However, some weaknesses are present regarding fault-tolerance due to a single point-of-failure i.e Traffic Control Center. Additionally, this Traffic Control Center is responsible for a relatively large area which might affect the system response-time to deliver in real-time traffic information to vehicle drivers. Finally, since it's necessary to install, manage and maintain a large number of roadside sensors this might become expensive.

Vehicle Information and Communication System (VICS) [4] is a digital data communication system used in Japan that promptly provides the latest necessary road traffic information to vehicle drivers via car navigation equipment. The highway administrators and the prefectural police are responsible for providing the traffic collected information to the Japan Road Traffic Information Center. This Center forwards this data to the VICS Center where other information like the availability on parking is collected. Then, the task of the VICS Center is to process and edit all information. It systematically connects all information from the roads and sends them from beacons set up on roads, using infrared rays on main trunk roads and radio waves (quasi-microwaves) on expressways. Thus, the road traffic information necessary by vehicle drivers is provided by radio beacons on expressways and optical beacons on major trunk roads. In addition, FM multiplex broadcasting via FM radio waves provides information on road traffic conditions. Now all information is available to vehicle drivers depending on the three different on-board equipments that supply different dept of information.

VICS has some strengths such as: high accuracy **w.r.t** current state of the traffic situation. However, some weaknesses are present regarding the requirement of extra-equipment attached-to vehicles which limits its availability to vehicle drivers that buy their specific equipment, the use of external equipment which might compromise driver's privacy and fault-tolerance due to a single point-of-failure i.e VICS Center.

Freeway Traffic Management System (COMPASS) [5] is a sophisticated advanced traffic management system run by the Ontario Ministry of Transportation (OTM) to regulate traffic flow on the 400-series highways. COMPASS uses electromagnetic vehicle loop detectors embedded in the pavement to measure volume, speed, and lane occupancy. This data is then transmitted to a central computer every 20 seconds using the latest in fiber optic communications technology and analyzed by operators, who also view the feeds of traffic cameras placed along the highways. Afterwards, Changeable Message Signs (CMS) display messages to vehicle drivers on the highways, advising them of upcoming collisions, closures, detours and traffic flow.

COMPASS has several strengths but the most notable one is the ability to use incident detection algorithms to infer changes in traffic conditions. In addition, COMPASS shares the same strengths with systems that use road-sensors and Traffic Control Centers in order to provide traffic information to vehicle drivers such as high accuracy, high scalability and no need for extra-equipment in vehicles. However, it shares also the same weaknesses such as a single point of failure and the need to install, manage and maintain a large number of road sensors/traffic cameras which might become costly.

Self-organizing Traffic Information System (SOTIS) [6] is a system based on self-organizing inter-vehicle communication. In contrast to the existing conventional traffic information systems, no central station that performs traffic analysis is required. Thus, each vehicle autonomously performs a traffic

analysis and broadcasts the results to other vehicles within transmission range which means that no fixed infrastructure is necessary. After the conclusion of this process, the traffic information is displayed on the in-vehicle navigation system.

SOTIS has several strengths such as: delivering traffic information on the surrounding area with low delay and a high level of detail, fault-tolerance since each vehicle is responsible for performing traffic analysis, and high scalability by adding extra vehicles to the system. However, some weaknesses are present regarding the requirement of extra-equipment attached-to vehicles (e.g. GPS) which might compromise driver's privacy.

2.8 Synthesis

In this chapter, we started by describing how ITS services are organized within three groups: **Traffic Safety**, **Traffic Efficiency** and **Value-Added Services**. We came to the conclusion that **RoadAhead** can be included in both Traffic Safety and Traffic Efficiency since its main focus is to deliver a broader vision of the traffic conditions in order to help vehicle drivers make better decisions in real-time.

After, we described and analysed three different ITS architectures stating their benefits and issues. We compared them with the requirements that RoadAhead must fulfill and found that V2I architecture meets almost all the requirements that we set ourselves failing only to achieve a fast response-time and least cost solution. In addition, we saw that V2V architecture meets some of our requirements (e.g. fast response-time and scalability) but due to problems related with traffic density (which affect system's accuracy) we consider it as an invalid option. Someone could say that V2V2I architecture should be our choice, since as well as V2V architecture is scalable and has a fast response-time but since it's based on V2I architecture it's also accurate. However, both V2V and V2V2I architectures present crucial flaws to achieve our goal which are not shared by the V2I architecture. These flaws are e.g. improper behavior when low traffic density is present, the fact that driver's privacy may be compromised, and requirement of extra equipment attached-to or built-in vehicles to operate correctly. Due to these facts, we concluded that V2I architecture is the most suitable to our system.

Later on, we identified and described detection mechanisms for vehicles on road which are grouped in : **Infrastructed-based detectors (intrusive and non-intrusive)** and **Vehicle-based detectors**. Since our system is required to be non-intrusive and available to all vehicle drivers we concluded that both intrusive and vehicle-based detectors can be discarded and only non-intrusive detectors are able to fulfill the above mentioned requirements. Thus, we decided that **RoadAhead** should use the Microwave Radar Sensor because when compared to its remaining competitors it is able to infer several essential traffic parameters with a lower overall cost.

Afterwards, we talked about the data analysis process and its importance to a system that uses huge amounts of data from different heterogeneous devices. We concluded that the ability to combine all the data collected by sensors or other devices in an appropriate manner is a key issue for information systems solution providers and a challenge for the data processing systems and methods.

Finally, we described proposed methods that can be used along with computational efforts to determine levels of congestion from traffic flow information, different systems that can be used to disseminate traffic information to vehicle drivers stating their strengths/weaknesses, and existing systems that have some similarities with **RoadAhead**.

Chapter 3

Architecture

This chapter addresses the architecture proposed for **RoadAhead** focusing on its main components: road sensors, local servers and light emitting diodes (LEDs), and how our solution fulfills the challenges that have been identified. Furthermore, we outline the major reasons for the decisions taken explaining them in more detail.

3.1 RoadAhead Architecture

As mentioned in this document before, **RoadAhead** uses a decentralized architecture, as depicted in Fig. 3.1, which is based on the existing Vehicle-to-Infrastructure (V2I) architecture.

As one can see, **RoadAhead** divides the existing road network into several well defined zones with variable size (e.g. $Zone_1 ... Zone_n$), each containing one or more roads, depending upon the traffic flow of each road. Moreover, each zone relies on three major components which together manage to deliver the current traffic situation in real-time, namely: road sensors, a local server and light emitting diodes (LEDs).

Regarding road sensors, they are deployed along the roads and responsible for the collection of traffic information i.e. vehicle instantaneous velocity which is forwarded to the correspondent local server. There, this information is used to infer the current traffic situation by applying a traffic and congestion estimation and forecasting method which evaluates and divides it into three distinct states: <Free Flow> when the traffic is normal and fluid, <Slightly Congested> when there are light vehicle stops, and <Congested> when the traffic has long stoppages. Finally, the local server translates the calculated traffic congestion which is delivered to LEDs that are disposed parallel to the road by using a color grade system constituted by three different colors: Green, Yellow and Red, where each color corresponds to a traffic congestion state, respectively <Free Flow>, <Slightly Congested>, and <Congested>.

3.1.1 Road Sensors

Road sensors are deployed along the roads and spaced with a fixed interval (e.g. 100 meters, 250 meters, 500 meters, etc..). They are responsible for the collection of traffic information through the

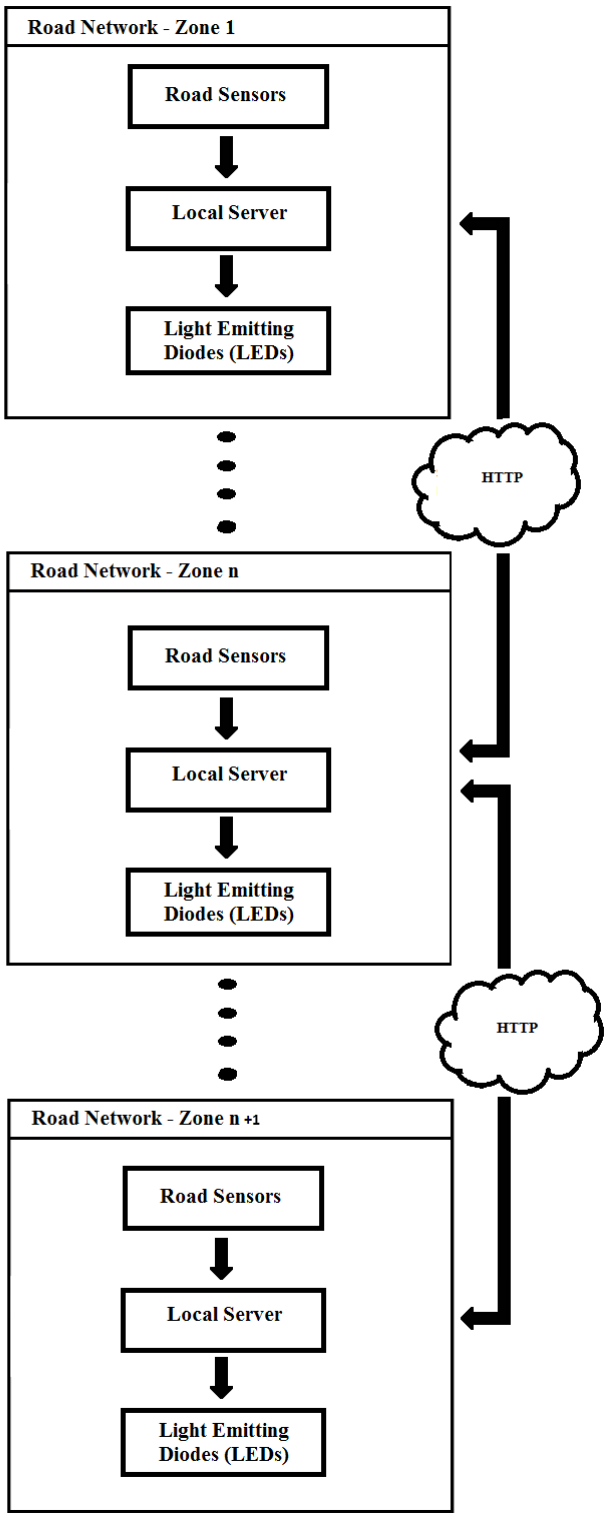


Figure 3.1: RoadAhead Architecture.

detection of vehicles that pass through them. Therefore, each time a vehicle is detected they forward the collected traffic information i.e. vehicle instantaneous velocity to the correspondent local server in order to be used as input by a traffic and congestion estimation and forecasting method.

Due to the requirement of our system to be non-intrusive to vehicle drivers we decided that from among all the presented road sensors the one that most suits **RoadAhead** is the microwave radar sensor because it has fewer limitations when compared to other road sensors available in the market. It requires minimal disruption of the traffic network for installation, operation and maintenance, has the best cost-benefit ratio among every road sensor that has been presented in this document and is capable of inferring essential traffic parameters (e.g. Vehicle instantaneous velocity, volume, etc..) which are very useful for the final classification of the traffic congestion. Finally, this road sensor doesn't gather any information that could undermine vehicle driver's privacy such as: position, vehicle licence plate, etc.. and doesn't require any extra equipment attached-to or built-in vehicles in order to reach the maximum number of vehicle drivers at a given road.

3.1.2 Local Servers

One of the most important components of **RoadAhead** and the main core of our system is the local server which is responsible for three important tasks:

1. Receive at a given time traffic information i.e. Vehicle instantaneous velocity provided by the road sensors assigned to its Zone.
2. Use the aformentioned information as input to a traffic and congestion estimation and forecasting method (explained in more detail later on Section 3.1.2.1) that infers and evaluates the current traffic situation by dividing it into three distinct states: <Free Flow> when the traffic is normal and fluid, <Slightly Congested> when there are light vehicle stops, and <Congested> when the traffic has long stoppages.
3. Disseminate the current traffic situation to vehicle drivers by using light emitting diodes displaced at the roadside.

3.1.2.1 Infering Traffic Congestion with a Simple Threshold Method

Relying on the traffic information provided by the road sensors, which requires minimal computational effort, as well as, minimal input data, a traffic congestion state is determined by applying a three-step procedure, based on the method proposed by Pattara-Atikom et al. [35], to the collected vehicle instantaneous velocities as follows:

1. Smoothening out the vehicle instantaneous velocities.
2. Classifiying and determining the initial congestion state.
3. Determining the final congestion state to be disseminated to vehicle drivers.

As mentioned before, the first step of the method is to smoothen out the vehicle instantaneous velocities and it's used because the instantaneous velocities collected by the road sensors are usually wildly fluctuated which makes it difficult to determine appropriate congestion states. Pattara-Atikom et al.

Time Period (t)	$V_{instantaneous}$ (km/h)
2	15
3	10
4	5
5	45
6	50
7	5
8	10
9	5
10	15
11	5
12	10
13	50
14	45

Table 3.1: Traffic information obtained by a road sensor and delivered to a local server at time period t

observed that the fluctuation behavior of vehicle velocity is similar to that of throughput of Transportation Control Protocol (TCP) flow of the Internet. Therefore, it's applied the same technique that TCP uses to smoothen out the traffic when calculating average throughput which is the **Exponential Weighted Moving Average Method**.

The formula used by this technique can be expressed mathematically as presented below:

$$V_{average}(t) = \begin{cases} V_{instantaneous}(t) & \text{if } t = 1 \\ \alpha \times V_{instantaneous}(t) + (1-\alpha) \times V_{average}(t-1) & \text{if } t > 1 \end{cases} \tag{3.1}$$

Where:

1. The coefficient α represents the weighted decrease degree with a factor between 0 and 1. A higher value α discards older vehicle instantaneous velocities faster.

2. $V_{instantaneous}(t)$ is the vehicle instantaneous velocity provided by a road sensor at a time period t.

3. $V_{average}(t)$ is the average velocity of a road sensor at any time period t based on the vehicle instantaneous velocities of different vehicles.

4. $V_{average}(1)$ is initially undefined. Due to this, we set $V_{average}(1) = V_{instantaneous}(1)$ which may be initialized in a number of different ways such as the average of the first four or five vehicle instantaneous velocities. The importance of the $V_{average}(1)$ initializations affect future results since the exponential weighted average depends on α. This means that smaller α values make the choice of $V_{average}(1)$ relatively more important than larger α values, since a higher α discards older vehicle instantaneous velocities faster.

In order to better understand how to smoothen out the vehicle instanteneous velocities, consider as an example Table 3.1, where is represented the vehicle instanteneous velocities detected by a road sensor on a road and sent to a local server at time period t. If we take into consideration the **Exponential Weighted Moving Average Method** (3.1) and time period $t = 2$ with $V_{instantaneous}(2) = 15\ km/h$,

Time Period (t)	$V_{instantaneous}$ (km/h)	$V_{average}$ (km/h)
2	15	38
3	10	28
4	5	20
5	45	29
6	50	36
7	5	25
8	10	20
9	5	15
10	15	15
11	5	12
12	10	11
13	50	25
14	45	32

Table 3.2: Traffic information obtained by a road sensor and delivered to a local server at time period t and $V_{average}(t)$ calculated with the Exponential Weighted Moving Average Method ; $V_{average}(1) = V_{instantaneous}(1) = 50\ km/h$ and $\alpha = 0,35$

$V_{average}(1) = V_{instantaneous}(1) = 50km/h$ and $\alpha = 0,35$ we obtain the following $V_{average}$ of the road sensor at that time period:

$$V_{average}(2) = 0,35 \times V_{instantaneous}(2) + (1 - 0,35) \times V_{average}(1)$$

$$\Downarrow$$

$$V_{average}(2) = 0,35 \times 15 + 0,65 \times 50$$

$$\Downarrow$$

$$V_{average}(2) = 38\ km/h \tag{3.2}$$

With this in mind, Table 3.2 presents the smoothening out of each vehicle instantaneous velocity contained in Table 3.1 after applying the **Exponential Weighted Moving Average Method** (3.1) with $V_{average}(1) = V_{instantaneous}(1) = 50\ km/h$ and $\alpha = 0,35$.

Now that we smoothen out the vehicle instantaneous velocities we proceed to the second step, where the aformentioned $V_{average}(t)$ is classified. However, in order to correctly classify it, it's necessary to know beforehand the type of road (e.g.) where **RoadAhead** is deployed, and more importantly, the road speed limit allowed by law which is usually the maximum speed. Thus, based on this information a simple threshold technique is used to classify $V_{average}(t)$ into three distinct congestion states:

<**Free Flow**>

when the road traffic is moving with normal velocity i.e. close to the road velocity limit and with small or very short slow-downs or pauses. For this congestion state, vehicle drivers are advised to maintain velocity.

<**Slightly Congested**>

when the road traffic is moving with moderate velocity i.e. close to the minimum road velocity

Time Period (t)	$V_{average}$ (km/h)	$Congestion_{InitialState}$
2	38	Free Flow
3	28	Slightly Congested
4	20	Slightly Congested
5	29	Slightly Congested
6	36	Free Flow
7	25	Slightly Congested
8	20	Slightly Congested
9	15	Congested
10	15	Congested
11	12	Congested
12	11	Congested
13	25	Slightly Congested
14	32	Free Flow

Table 3.3: Classifying and Determining Initial Congestion State ($\beta = 15\ km\backslash h$; $\gamma = 30\ km\backslash h$)

limit and with frequent and long pauses. For this congestion state, vehicle drivers are advised to moderate velocity.

$<$**Congested**$>$

when the road traffic is moving with low velocity i.e. forced to a complete stop or move very slowly for an extended period of time due to incidents (e.g. road construction, accidents). For this congestion state, vehicle drivers are advised to be careful and reduce velocity.

This simple threshold technique can be expressed mathematically as shown below:

$$\begin{aligned} V_{average}(t) \geq \gamma &\implies <Free\ Flow> \\ \beta < V_{average}(t) < \gamma &\implies <Slightly\ Congested> \\ V_{average}(t) \leq \beta &\implies <Congested> \end{aligned} \tag{3.3}$$

Where:

1. $V_{average}(t)$ is the average velocity of a road sensor at any time period t.

2. β and γ are static velocity boundaries (e.g. $\beta = 15\ km\backslash h$; $\gamma = 30\ km\backslash h$) defined by **RoadAhead** in order to calculate the different states of traffic congestion.

In order to grasp how the second step is applied, consider as an example Table 3.2, in which the last column contains the average velocity of a road sensor at time period t. Thus, if we apply the simple threshold technique to time period $t = 2$ with $V_{average}(2) = 38\ km/h$, $\beta = 15\ km\backslash h$ and $\gamma = 30\ km\backslash h$, we obtain the following congestion state for that time period:

Time Period (t)	$V_{average}$ (km/h)	$Congestion_{InitialState}$	$Congestion_{FinalState}$	Led Color
2	38	Free Flow	Free Flow	Green
3	28	Slightly Congested	Free Flow	Green
4	20	Slightly Congested	Slightly Congested	Yellow
5	29	Slightly Congested	Slightly Congested	Yellow
6	36	Free Flow	Slightly Congested	Yellow
7	25	Slightly Congested	Slightly Congested	Yellow
8	20	Slightly Congested	Slightly Congested	Yellow
9	15	Congested	Slightly Congested	Yellow
10	15	Congested	Congested	Red
11	12	Congested	Congested	Red
12	11	Congested	Congested	Red
13	25	Slightly Congested	Congested	Red
14	32	Free Flow	Congested	Red

Table 3.4: Determining Final Congestion State

$$
\begin{aligned}
V_{average}(2) \geq 30\ km/h &\implies <Free\ Flow> \\
15\ km/h < V_{average}(2) < 30\ km/h &\implies <Slightly\ Congested> \\
V_{average}(2) \leq 15\ km/h &\implies <Congested> \\
&\Downarrow \\
38\ km/h \geq 30\ km/h &\implies <Free\ Flow> \\
15\ km/h < 38\ km/h < 30\ km/h &\implies <Slightly\ Congested> \\
38\ km/h \leq 15\ km/h &\implies <Congested> \\
&\Downarrow \\
38\ km/h \geq 30\ km/h &\implies <Free\ Flow>
\end{aligned}
\tag{3.4}
$$

With this in mind, Table 3.3 shows the classification of the initial congestion state after applying the simple threshold technique to the average velocities contained in Table 3.2 with $\beta = 15\ km\backslash h$ and $\gamma = 30\ km\backslash h$. However, by observing Table 3.3, one can conclude that sometimes when receiving consecutive instantaneous velocities the congestion state can vary abruptly which force the average velocity of the road sensor to suddently change (e.g. from $t = 5$ to $t = 7$). Thus, any congestion event that behaves like this is considered as a transient event and should not be counted or reported as a new congestion state. Therefore, congestion events should last at least more than a few received instantaneous velocities.

Due to this, we proceed to apply the third-step in order to classify the final congestion state by using a formula that determines the transition between congestion states as follows:

$$Congestion_{FinalState}(t) = \begin{cases} Congestion_{InitialState}(t) & \text{if } t = 1 \\ Congestion_{InitialState}(t) & \text{if } t > 1, \\ & Congestion_{InitalState}(t-1) = Congestion_{InitalState}(t), \\ & Congestion_{FinalState}(t-1) \neq Congestion_{InitialState}(t) \\ Congestion_{FinalState}(t-1) & \text{otherwise} \end{cases} \tag{3.5}$$

Where:

1. $Congestion_{InitialState}(t)$ is the initial congestion state after applying the simple threshold technique to the average velocity at time period t.

2. $Congestion_{FinalState}(t)$ is the final congestion state after applying the third-step of the method at time period t.

In order to understand how to obtain the final congestion state, consider as an example Table 3.3, in which the last column contains the initial congestion state at time period t. Thus, if we apply the above formula to time period $t = 2$, we obtain the following final congestion state for that time period:

$$Congestion_{FinalState}(2) = \begin{cases} Congestion_{InitialState}(2) & \text{if } 2 = 1 \\ Congestion_{InitialState}(2) & \text{if } 2 > 1, \\ & Congestion_{InitalState}(1) = Congestion_{InitalState}(2), \\ & Congestion_{FinalState}(1) \neq Congestion_{InitialState}(2) \\ Congestion_{FinalState}(1) & \text{otherwise} \end{cases}$$

$$\Downarrow$$

$$Congestion_{FinalState}(2) = \begin{cases} < Slightly\ Congested > & \text{if } 2 > 1, \\ & < Free\ Flow > = < Slightly\ Congested >, \\ & < Free\ Flow > \neq < Slightly\ Congested > \\ < Free\ Flow > & \text{otherwise} \end{cases}$$

$$\Downarrow$$

$$Congestion_{FinalState}(2) = < Free\ Flow > \tag{3.6}$$

To conclude, Table 3.4 shows the final congestion states after applying the third step of the procedure to the values contained in Table 3.3.

Time Period (t)	$Congestion_{FinalState}$	Led Color
2	Free Flow	Green
3	Free Flow	Green
4	Slightly Congested	Yellow
5	Slightly Congested	Yellow
6	Slightly Congested	Yellow
7	Slightly Congested	Yellow
8	Slightly Congested	Yellow
9	Slightly Congested	Yellow
10	Congested	Red
11	Congested	Red
12	Congested	Red
13	Congested	Red
14	Congested	Red

Table 3.5: Determining LED color based on final congestion state

3.1.3 Light Emitting Diodes (LEDs)

After determining the correspondent traffic congestion state, it's necessary to forward this information to the intended receivers, which are those who are interested in it (e.g. vehicle drivers, road users, ITS personnel).

In this way, our system uses the local servers to disseminate this information to light emitting diodes (LEDs) assigned to each of the existing zones (e.g. $Zone_1 ... Zone_n$) which have the ability to visually represent the congestion state of a certain road. These LEDs are deployed on the ground and parallel to the road with a fixed interval between them (e.g. 1 meter, 5 meters, etc..) and are constituted by three different colors that change according to the traffic situation. Thus, our system associates each traffic congestion state to the corresponding color as shown in Table 3.5 based on the following rules:

$$\begin{aligned} <Free\ Flow> &\iff GREEN \\ <Slightly\ Congested> &\iff YELLOW \\ <Congested> &\iff RED \end{aligned} \tag{3.7}$$

To conclude, we decided to use light emitting diodes (LEDs) because none of the tools presented in this document can represent reliable information in real-time to vehicle drivers without some sort of delay, extra equipment attached-to or built-in vehicles, or causing distraction to vehicle drivers.

3.2 Synthesis

In this chapter we started by describing the decentralized architecture proposed for **RoadAhead** based on the existing Vehicle-to-Infrastructure (V2I) architecture. Thus, we saw that **RoadAhead** divides the existing road network into several well defined zones with variable size (e.g. $Zone_1 ... Zone_n$), each containing one or more roads, depending upon the traffic flow of each road. In addition, we stated

that each zone relies on three major components which together manage to deliver the current traffic situation in real-time, namely: road sensors, a local server and light emitting diodes (LEDs).

Regarding road sensors, we observed that from among all the presented road sensors the one that most suits **RoadAhead** is the microwave radar sensor. Thus, this is due to the requirement of **RoadAhead** to be non-intrusive to vehicle drivers and because this road sensor has fewer limitations when compared to other road sensors available in the market. Hence, this type of road sensor is deployed along the roads and responsible for the collection of traffic information i.e. vehicle instantaneous velocity which is forwarded to the correspondent local server. There, a traffic congestion state is determined by applying a three-step procedure, based on the method proposed by Pattara-Atikom et al. [35], to the collected vehicle instantaneous velocities, where there are three possible traffic congestion states to be infered, <Free Flow> when the traffic is normal and fluid, <Slightly Congested> when there are light vehicle stops, and <Congested> when the traffic has long stoppages.

Finally, the local server translates the calculated traffic congestion which is delivered to LEDs that are disposed parallel to the road by using a color grade system constituted by three different colors: Green, Yellow and Red, where each color corresponds to a traffic congestion state, respectively <Free Flow>, <Slightly Congested>, and <Congested>.

To conclude, we decided to use light emitting diodes (LEDs) because none of the tools presented in this document can represent reliable information in real-time to vehicle drivers without some sort of delay, extra equipment attached-to or built-in vehicles, or causing distraction to vehicle drivers.

Chapter 4

Implementation

This chapter focuses on the most relevant details regarding the implementation of **RoadAhead** namely, its main components: road sensors, local servers and light emitting diodes (LEDs). Furthermore, we describe the major reasons for the implementation decisions taken explaining them in more detail.

4.1 RoadAhead Implementation

In order to evaluate our system and due to the prohibitive cost of deploying a large sensor and light emitting diode network we decided to follow the conventional approach used by most researches that rely on simulation tools. However, instead of using existing simulation tools we decided to implement our own due to the following reasons:

1. Flexibility
   We are not restricted to a particular development environment, which means that we can choose what development tools and languages to use such as programming language, integrated development environment, different or additional versions of libraries and support software, etc..

2. Complexity
   Implementing our own simulation tool based on an architecture with well defined requirements it's much easier than changing an existing simulation tool where we need to figure out what to modify in order to behave as expected.

With this said, Fig. 4.1 presents our simulation tool, which is an overview of **RoadAhead** where each main component is depicted as a distinct abstract layer i.e. road sensors are represented as client layer, local server as server layer, and finally, light emitting diodes as graphical user interface layer. Thus, in the following sections we explain in more detail how we implemented each of these layers and how they interact with each other.

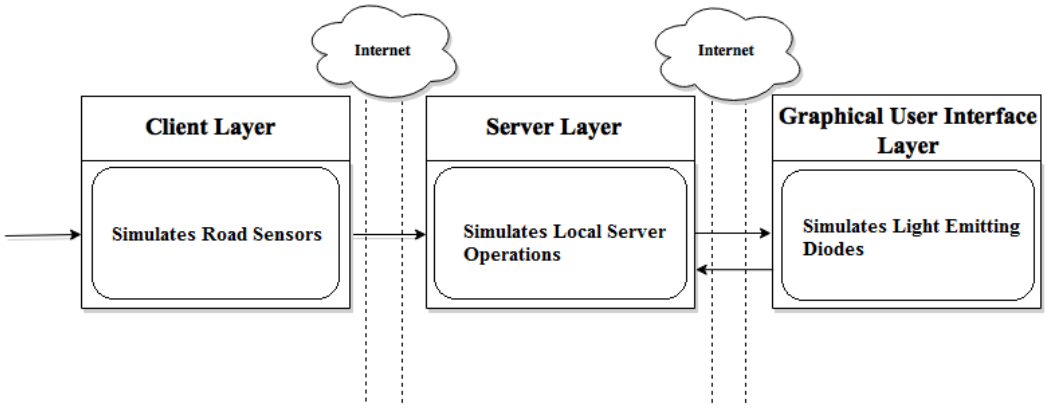


Figure 4.1: RoadAhead Implementation Layers

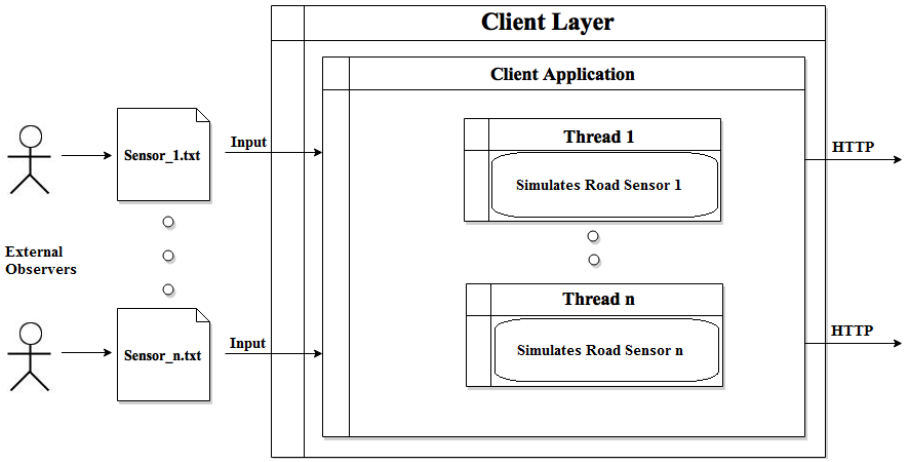


Figure 4.2: Client Layer

4.1.1 Client Layer

In order to simulate the same functioning of road sensors as in the real world we implemented a stand-alone client-based application written in Java™ language using Java Platform, Standard Edition that emulates the road sensor network.

Fig. 4.2 presents a more detailed representation of the client layer where our client application is included. As one can see the client application receives as input several text files, each of them representing a given road sensor, which are manually created by external observers that visually track the roads at some well defined points (e.g. each external observer is separated by 100 meters, 250 meters, 500 meters, etc.. simulating in this way the same placement of road sensors as in the real world). Thus, with the help of these external observers each file is populated with traffic information, as shown in Fig. 4.3, with the following data representation:

Line 1: Zone = $Zone_n$ Sensor = $Sensor_n$ Velocity = $V_{instantaneous}(t)$

Line 2: Zone = $Zone_n$ Sensor = $Sensor_n$ Velocity = $V_{instantaneous}(t+1)$

...

Where:

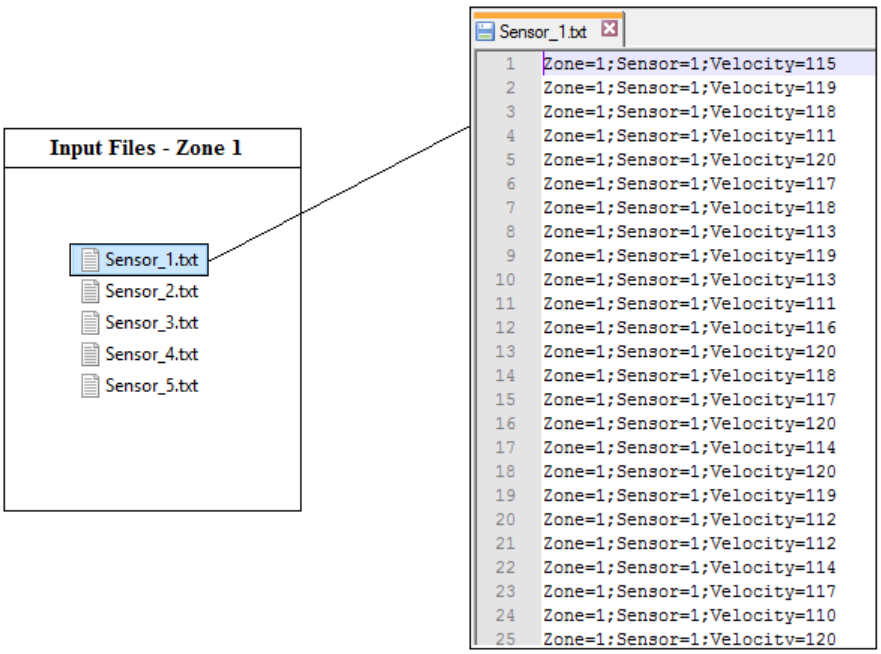


Figure 4.3: Traffic information obtained by an external observer ($Zone_1$, $Sensor_1$)

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:sch="http://zoneserver.com/example/schemas">
   <soapenv:Header/>
   <soapenv:Body>
      <sch:VelocityRequest>
         <sch:Zone>1</sch:Zone>
         <sch:Sensor>1</sch:Sensor>
         <sch:Velocity>115</sch:Velocity>
      </sch:VelocityRequest>
   </soapenv:Body>
</soapenv:Envelope>
```

Figure 4.4: SOAP request message

1. $Zone_n$ is the numerical representation of a zone.

2. $Sensor_n$ is the numerical representation of a sensor.

3. $V_{instantaneous}(t)$ is the vehicle instantaneous velocity provided by a road sensor at a time period t.

Subsequently, the client application emulates the road sensor network by associating these input files to threads, namely a thread per input file where each thread represents a particular road sensor. Thus, each thread simulates the detection of vehicles passing through a road sensor by reading the traffic information values associated to its input file where each line represents the detection of a vehicle.

Finally, this traffic information needs to be forwarded to the correspondent local server in the server layer. In this way, each time a thread reads a line it communicates its values through the invocation of a webservice operation, as shown in Fig. 4.4, implemented in the server layer and represented by an XML-based protocol, referred to as SOAP that relies on application layer protocols, most notably Hypertext Transfer Protocol (HTTP), for message negotiation and transmission (explained in more detail later on 4.1.2).

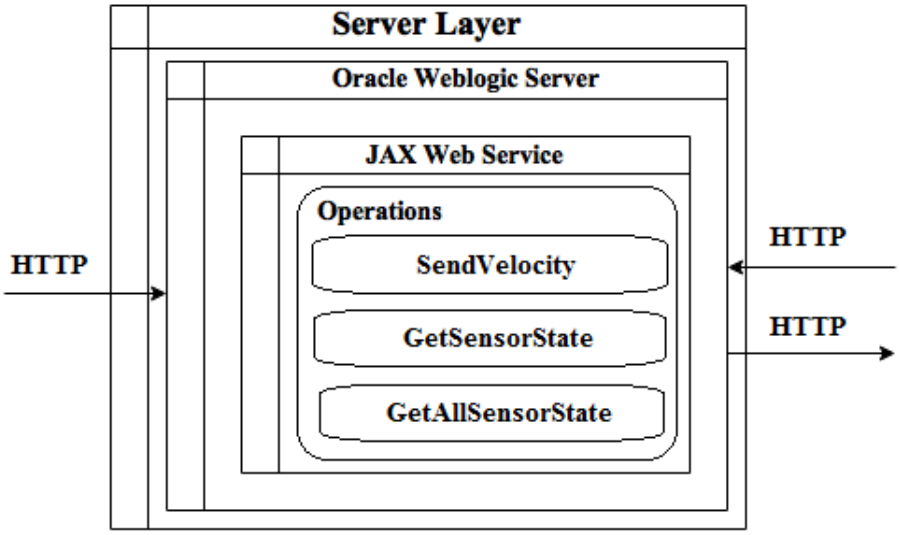


Figure 4.5: Server Layer

4.1.2 Server Layer

In order to implement the logic behind the local server, as described in Chapter 3, and consequently, the communication between the different layers, as presented in Fig. 4.5, we decided to use a solution widely used in systems integration and communication between different applications called **Oracle Weblogic Server WebServices**, which is one of the core components of **Oracle Weblogic Server**, a Java Platform, Enterprise Edition application server currently developed by Oracle Corporation.

Oracle Weblogic Server WebServices can be implemented by using the Java API for XML-based webservices (JAX-WS) which is a standard-based API for coding, assembling, and deploying Java web-services. Additionally, in JAX-WS an invocation of a webservice operation is represented by an XML-based protocol, such as SOAP. The SOAP specification defines the envelope structure, encoding rules, and conventions for representing webservice invocations and responses which are transmitted as SOAP messages (XML files) over HTTP.

With this in mind, in order to implement our webservice we started by using a top-down model approach, also known as contract-first, where a WSDL document is written first. This document is a public contract that specifies what a webservice looks like, such as the list of supported operations, the signature and shape of each operation, the protocols and transports that can be used when invoking the operations, and the XML Schema data types that are used when transporting the data. Thus, in Listing 4.1 is presented the definition of our WSDL document which specifies our webservice. As one can see, the webservice exposes three operations to the outside as follows:

1. **SendVelocity**, this operation is responsible by the reception of the instantaneous velocities
2. **GetSensorState**, this operation returns the traffic congestion state of a given road sensor
3. **GetAllSensorState**, this operation returns the traffic congestion state of the road network

Moreover, based on this WSDL document a code generating tool provided by **Orable Weblogic**

Server automatically generates the artifacts that implement the webservice, which are then modified in order to implement the business requirements, and subsequently deployed to the application server.

Listing 4.1: WSDL - RoadAhead Webservice

```
<wsdl:definitions xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
                  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
                  xmlns:schema="http://zoneserver.com/example/schemas"
                  xmlns:tns="http://zoneserver.com/example/definitions"
                  targetNamespace="http://zoneserver.com/example/definitions">
    <wsdl:types>
        <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
            <xsd:import namespace="http://zoneserver.com/example/schemas"
                schemaLocation="sensor.xsd"/>
        </xsd:schema>
    </wsdl:types>

    <!-- Messages -->
    <wsdl:message name="VelocityRequest">
        <wsdl:part element="schema:VelocityRequest" name="VelocityRequest"/>
    </wsdl:message>

    <wsdl:message name="GetSensorStateRequest">
        <wsdl:part element="schema:GetSensorStateRequest" name="GetSensorStateRequest"/>
    </wsdl:message>

    <wsdl:message name="GetSensorStateResponse">
        <wsdl:part element="schema:GetSensorStateResponse" name="GetSensorStateResponse"/>
    </wsdl:message>

    <wsdl:message name="GetAllSensorStateRequest">
        <wsdl:part element="schema:GetAllSensorStateRequest" name="GetAllSensorStateRequest"/>
    </wsdl:message>

    <wsdl:message name="GetAllSensorStateResponse">
        <wsdl:part element="schema:GetAllSensorStateResponse" name="GetAllSensorStateResponse"
    />
    </wsdl:message>

    <!-- Port Type -->
    <wsdl:portType name="VelocityPort">
        <wsdl:operation name="SendVelocity">
            <wsdl:input message="tns:VelocityRequest" name="VelocityRequest"/>
        </wsdl:operation>
        <wsdl:operation name="GetSensorState">
          <wsdl:input message="tns:GetSensorStateRequest" name="GetSensorStateRequest"/>
            <wsdl:output message="tns:GetSensorStateResponse" name="GetSensorStateResponse"/>
        </wsdl:operation>

        <wsdl:operation name="GetAllSensorState">
```

```
            <wsdl:input message="tns:GetAllSensorStateRequest" name="GetAllSensorStateRequest"
    />
            <wsdl:output message="tns:GetAllSensorStateResponse" name="
    GetAllSensorStateResponse"/>
        </wsdl:operation>
    </wsdl:portType>

    <!-- Binding -->
    <wsdl:binding name="VelocityBinding" type="tns:VelocityPort">
        <soap:binding style="document"
            transport="http://schemas.xmlsoap.org/soap/http"/>
        <wsdl:operation name="SendVelocity">
            <soap:operation soapAction="http://zoneserver.com/SendVelocity"/>
            <wsdl:input name="VelocityRequest">
                <soap:body use="literal"/>
            </wsdl:input>
        </wsdl:operation>

        <wsdl:operation name="GetSensorState">
            <soap:operation soapAction="http://zoneserver.com/GetSensorVelocity"/>
            <wsdl:input name="GetSensorStateRequest">
                <soap:body use="literal"/>
            </wsdl:input>
            <wsdl:output name="GetSensorStateResponse">
                <soap:body use="literal"/>
            </wsdl:output>
        </wsdl:operation>

        <wsdl:operation name="GetAllSensorState">
            <soap:operation soapAction="http://zoneserver.com/GetAllSensorState"/>
            <wsdl:input name="GetAllSensorStateRequest">
                <soap:body use="literal"/>
            </wsdl:input>
            <wsdl:output name="GetAllSensorStateResponse">
                <soap:body use="literal"/>
            </wsdl:output>
        </wsdl:operation>
    </wsdl:binding>

    <!-- Service -->
    <wsdl:service name="VelocityService">
        <wsdl:port binding="tns:VelocityBinding" name="VelocityPort">
            <soap:address location="http://localhost:7001/JavaZoneServer/VelocityService/"/>
        </wsdl:port>
    </wsdl:service>
</wsdl:definitions>
```

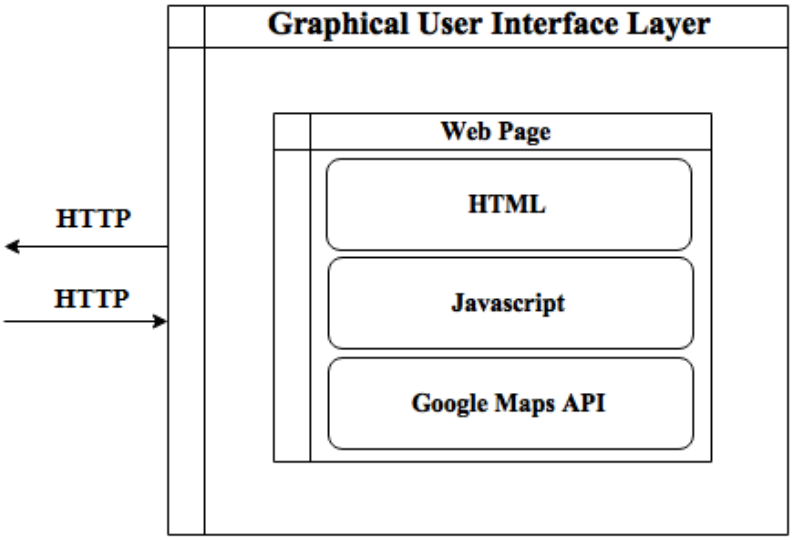


Figure 4.6: Graphical User Interface Layer

4.1.3 Graphical User Interface Layer

Due to the prohibitive cost of deploying a light emitting diode network, and in order to provide a similar experience as the one that we would have if light emitting diodes were displaced on the roadside, we decided to implement a graphical user interface that shows in real-time an overview of several road sections and the changes that occur in each one of them after a certain time.

Therefore, in order to implement the graphical user interface layer as shown in Fig. 4.6 we decided to develop a web page written in HyperText Markup Language, commonly referred to as HTML, and make it available to everyone through the Internet by the means of a web browser (e.g. Internet Explorer). In addition, HTML can embed scripts written in languages such as Javascript which affect the behavior of HTML web pages. Thus, we use Javascript and the help of the Google Maps API which lets us customize in a more refined way the default Google Map, and transform it into our graphical user interface as depicted in Fig. 4.7.

Figure 4.7: Graphical User Interface

By observing the figure, one can see that the default Google Map was modified by introducing six new icons representing our sensors, each one associated to a road section with a unique path. Moreover, each path can display three different colors that represent the congestion state of that particular road section at a given time. Thus, to obtain this visual effect we present in Listing 4.2 a portion of the implemented code that allows for the creation of a custom map and the dynamic draw of a particular path.

At first, we start by loading the Javascript library associated with the Google Maps API by adding it with a <script> tag (line 14 and 15). After this, we implement a function called e.g. initialize() where we create an object that contains the properties of the custom map (line 98 to 106). At this point, we are ready to create a custom map container with a element that holds the map and a custom map object that creates inside this container a custom map with the properties defined above (line 108).

Afterwards, we modify this map with the help of the Google Maps API in order to customize it with our road sensors and the correspondent road sections. Therefore, we use markers that represent single locations on a custom map which specify in this case the position of a road sensor on the surface of the earth by using GPS coordinates (line 111). Moreover, we use polylines to represent our road sections which are a series of straight lines on a custom map with variable length and color, and defined with a group of GPS coordinates (line 18 to 28).

However, for the time being, each road section displays irrelevant information because our graphical user interface doesn't communicate with the server layer yet. Therefore, in order to present the congestion state of each road section in real-time we use a method called setInterval() which calls a function i.e. getAllSensorsState() at specified intervals e.g. 1000ms (line 185). Thus, this function creates a SOAP request that is sent to the webservice, implemented in the server layer, in order to obtain information regarding each road section. Later on, the webservice responds with the congestion state of each road section which is used in runtime to change the color stroke of each polyline. Finally, and after everything is defined we add an event listener that executes the initialize() function in order to display our custom map everytime the page loads.

Listing 4.2: GUI - Custom Map creation

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4    <meta name="viewport" content="initial-scale=1.0, user-scalable=no">
5    <meta charset="utf-8">
6    <title>Traffic Congestion Interface</title>
7  <style>
8        html, body, #map-canvas {
9          height: 100%;
10         margin: 0px;
11         padding: 0px
12       }
13 </style>
14 <script type="text/javascript" src="https://maps.googleapis.com/maps/api/js?v=3.exp&sensor=
     false"></script>
```

```
<script type="text/javascript" src="http://google-maps-utility-library v3.googlecode.com/svn/trunk/infobox/src/infobox.js"></script>
<script>

var Zone1Sensor1Coordinates = [ new google.maps.LatLng(38.724911,-9.194101), new google.maps.LatLng(38.721914,-9.202555)];

{...}

  var Zone1Sensor1Path = new google.maps.Polyline({
    path: Zone1Sensor1Coordinates,
    geodesic: true,
    strokeColor: '#008000',
    strokeOpacity: 1.0,
    strokeWeight: 6
  });

{...}

function getSensorState(zone,sensor) {

  var xmlhttp = new XMLHttpRequest();
  xmlhttp.open('POST', 'http://localhost:7001/VelocityService', false);

          var sr = '<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" ' +
        'xmlns:sch="http://zoneserver.com/example/schemas"> ' +
             '<soapenv:Body> ' +
              '<sch:GetSensorStateRequest> ' +
                '<sch:Zone>'+zone+'</sch:Zone> ' +
                '<sch:Sensor>'+sensor+'</sch:Sensor> ' +
              '</sch:GetSensorStateRequest> '+
             '</soapenv:Body> ' +
          '</soapenv:Envelope> ';


            xmlhttp.setRequestHeader('SOAPAction', '"http://zoneserver.com/GetSensorState"');
            xmlhttp.setRequestHeader('Content-Type', 'text/xml; charset=UTF-8');
            xmlhttp.send(sr);
            xmlDoc = xmlhttp.responseXML;
            WS = xmlDoc.getElementsByTagName("GetSensorStateResponse");
            var a = WS[0].childNodes[0].childNodes[0].nodeValue;
            var b = WS[0].childNodes[1].childNodes[0].nodeValue;
            var c = WS[0].childNodes[2].childNodes[0].nodeValue;
            var d = WS[0].childNodes[3].childNodes[0].nodeValue;
            var content = '<div id="content" style="max-width: 150px; line-height: normal; white-space: nowrap; overflow: auto;">' + '<div id="bodyContent">'+ '<b>Zone:</b> '+a+'</br>'+ '<b>Sensor:</b> '+b+'</br>'+
                    '<b>Avg Velocity:</b> '+c+'</br>'+ '<b>Last Velocity:</b> '+d+'
```

```
    </br>'+
                '</div>'+
                '</div>';
  return content;
  }

 function getAllSensorsState() {

            var xmlhttp = new XMLHttpRequest();
            xmlhttp.open('POST', 'http://localhost:7001/VelocityService', true);

            var sr = '<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/
    envelope/" ' +
        'xmlns:sch="http://zoneserver.com/example/schemas"> ' +
             '<soapenv:Body> ' +
              '<sch:GetAllSensorStateRequest> ' +
              '</sch:GetAllSensorStateRequest> '+
             '</soapenv:Body> ' +
          '</soapenv:Envelope> ';
  xmlhttp.setRequestHeader('SOAPAction', '"http://zoneserver.com/GetAllSensorState"');
  xmlhttp.setRequestHeader('Content-Type', 'text/xml; charset=UTF-8');
  xmlhttp.send(sr);

  xmlhttp.onreadystatechange=function(){

    if (xmlhttp.readyState==4 && xmlhttp.status==200){
      xmlDoc = xmlhttp.responseXML;
      WS = xmlDoc.getElementsByTagName("GetAllSensorStateResponse");
      for(i=0; i < WS[0].childNodes.length; i++){
        var zone = WS[0].childNodes[i].childNodes[0].childNodes[0].nodeValue;
        var sensor = WS[0].childNodes[i].childNodes[1].childNodes[0].nodeValue;
        var level = WS[0].childNodes[i].childNodes[2].childNodes[0].nodeValue;
        var state = 'Zone'+zone+'Sensor'+sensor+'Path.setOptions({strokeColor: "'+level+'"})';
        eval(state);
      }
    }
  }
}

function initialize() {

var mapOptions = {
    zoom: 14,
    center: new google.maps.LatLng(38.716542,-9.222508),
    mapTypeId: google.maps.MapTypeId.TERRAIN,
  disableDefaultUI: true,
  draggable: true,
  scrollwheel: true,
  disableDoubleClickZoom: true
```

```
  };

var map = new google.maps.Map(document.getElementById('map-canvas'),
      mapOptions);

var SensorMarkerLang1 = new google.maps.LatLng(38.721818,-9.202478);

{...}

var SensorMarker1 = new google.maps.Marker({
      position: SensorMarkerLang1,
      map: map,
      title: 'Sensor 1',
    icon : '\icon.png'
 });

{...}

Zone1Sensor1Path.setMap(map);

{...}

var boxTextSensor1 = document.createElement("div");
boxTextSensor1.style.cssText = "border:2px solid black;"+
                "margin-top: 8px;"+
                "background:#333;"+
                "color:#FFF;"+
                "font-family:Arial, Helvetica, sans-serif;"+
                "font-size:12px;"+
                "padding: .5em 1em;"+
                "-webkit-border-radius: 4px;"+
                "-moz-border-radius: 4px;"+
                "border-radius: 4px;"+
                "text-shadow:0 -1px #000000;"+
                "-webkit-box-shadow: 0 0  8px #000;"+
                "box-shadow: 0 0 8px #000;";

{...}

var myOptionsSensor1 = {
       content: boxTextSensor1
      ,disableAutoPan: false
      ,maxWidth: 0
      ,pixelOffset: new google.maps.Size(-70,-125)
      ,zIndex: null
      ,boxStyle: {
        opacity: 0.99
        ,width: "150px"
       }
```

```
155        ,closeBoxMargin: "10px 2px 2px 2px"
156        ,closeBoxURL: "http://www.google.com/intl/en_us/mapfiles/close.gif"
157        ,infoBoxClearance: new google.maps.Size(1, 1)
158        ,isHidden: false
159        ,pane: "floatPane"
160        ,enableEventPropagation: false
161      };
162
163 {...}
164
165 var ibSensor1 = new InfoBox(myOptionsSensor1);
166
167 {...}
168
169 setInterval(function() { contentString= getSensorState(1,1);
170                 boxTextSensor1.innerHTML=contentString;
171                 if(!ibSensor1.getVisible())
172                 ibSensor1.open(map,SensorMarker1);},1000);
173 {...}
174
175 setInterval(function(){getAllSensorsState()},1000);
176
177 }
178 google.maps.event.addDomListener(window, 'load', initialize);
179
180   </script>
181   </head>
182   <body>
183     <div id="map-canvas" style='width:auto;'></div>
184 </body>
185 </html>
```

4.2 Synthesis

In this chapter we focused on describing the most relevant details regarding the implementation of **RoadAhead**, namely, its main components: road sensors, local server and light emitting diodes (LEDs). Thus, we started by explaining that due to the prohibitive cost of deploying a large sensor and light emitting diode network we followed the conventional approach used by most researches that rely on simulation tools. However, instead of using existing simulation tools we implemented our own due to the following reasons: **Flexibility** and **Complexity**.

Afterwards, we described the implementation layers of our simulation tool, namely: client, server, and graphical interface layer, and how they interact with each other, where each one simulates a different **RoadAhead** component, respectively, road sensors, local server and light emitting diodes (LEDs).

Regarding the client layer, we implemented a stand-alone client-based application written in Java™ language using Java Platform, Standard Edition that emulates the road sensor network by receiving

as input several text files, each one representative of a given road sensor and containing road traffic information i.e. instantaneous velocities to be transmitted to the server layer.

Concerning the server layer, and in order to implement the business logic behind the local server, we used a solution widely used in systems integration and communication between different applications called **Oracle Weblogic Server WebServices**, which is one of the core components of **Oracle Weblogic Server**, a Java Platform, Enterprise Edition application server currently developed by Oracle Corporation.

Finally, we presented our graphical interface layer that shows in real-time an overview of several road sections and the changes that occur in each one of them after a certain time. To do so, we developed a web page written in HyperText Markup Language, commonly referred to as HTML, which integrates Javascript and the Google Maps API.

Chapter 5

Evaluation

This chapter addresses the evaluation carried for **RoadAhead** by using the simulation tool described in chapter 4. In this way, we aim at determining if **RoadAhead** can improve the overall vehicle drivers experience if its components, i.e. road sensors, local server, and LEDs, are adequately positioned in the roadway. Moreover, an adequate positioning of these components might reduce the investment needed to deploy a complete road network infrastructure. Thus, the cost reduction along with the potential to provide reliable information to vehicle drivers in real-time is the main motivation for the simulations presented in this chapter.

5.1 Simulation Goals

The simulations presented in this chapter aim at optimizing the conditions in which **RoadAhead** should operate, and more importantly, the adequate positioning of its components in order to provide an accurate and reliable state of the current road traffic conditions. To do so, we vary three conditions of the system which we think appropriate in order to understand how it behaves, namely:

- **Road sensor distance** - We selected three distances between road sensors, 250 meters, 500 meters, and finally 1000 meters, in order to study the reaction time and accuracy of **RoadAhead**.
- **Simple threshold method static parameters** - The simple threshold method described in Chapter 3 relies on four static parameters, α, a weight given to each velocity where a higher α discards older vehicle instantaneous velocities faster, $V_{average}(1)$, which is initially undefined, and β and γ, static velocity boundaries defined based upon the maximum road velocity limits. Thus, we selected four different values of α, $\alpha = [0.25; 0.50; 0.75; 1]$, and $V_{average}(1)$, β and γ defined later based on the simulation setup.
- **Instantaneous Velocity** - Variation of the input instanteneous velocity in each road sensor ($1 - 120\ Km/h$).

With that said, Table 5.1 shows a synthesis of the previous conditions which are used in tree different scenarios:

Road Sensor Distance (m)	Number of Road Sensors	Static Parameter (α)	$V_{instantaneous}(km/h)$
250	40	0.25	[1-120]
		0.50	
		0.75	
		1	
500	20	0.25	
1000	10	0.25	

Table 5.1: RoadAhead conditions used by each of the scenarios

- **Free Flow**, when the road traffic is fluid and moving with velocity close to the maximum road velocity limit.
- **Slighty Congested**, when the road traffic is fluid but it starts to decrease until it gets near the minimum road velocity limit.
- **Emergency/Congested**, when the road traffic is fluid but it is forced to move very slowly for an extended period of time or to a complete stop due to an road accident.

5.2 Simulation Setup

In order to analyse the results of the previous described scenarios, we performed several tests that aim at simulating the constant road traffic in a highway, by using the simulation tool described in chapter 4. Thus, as presented in Fig. 5.1, we selected as example the Portuguese A5 highway (Lisboa-Cascais), more specifically, a 10 km long road section. However, since these tests occur in a simulation environment, we considered only a flat, one lane, straight road, with no traffic lights nor intersections, and with direction Cascais-Lisboa.

Regarding the input data needed to conduct these tests, namely the vehicles instantaneous velocities received by a road sensor, several input files were created for this purpose with random instantaneous velocities spaced with time intervals $\Delta t = 1s$, being each of these files representative of the scenario that we want to simulate. Therefore, these files were created with the following instantaneous velocities for each scenario:

- **Scenario I - Free Flow**, between 100 and 120 km/h.
- **Scenario II - Slighty Congested**, between 100 and 120 km/h but gradually decreasing down to 50 km/h.
- **Scenario III - Emergency/Congested**, between 100 and 120 km/h but drastically decreasing down to 10 km/h.

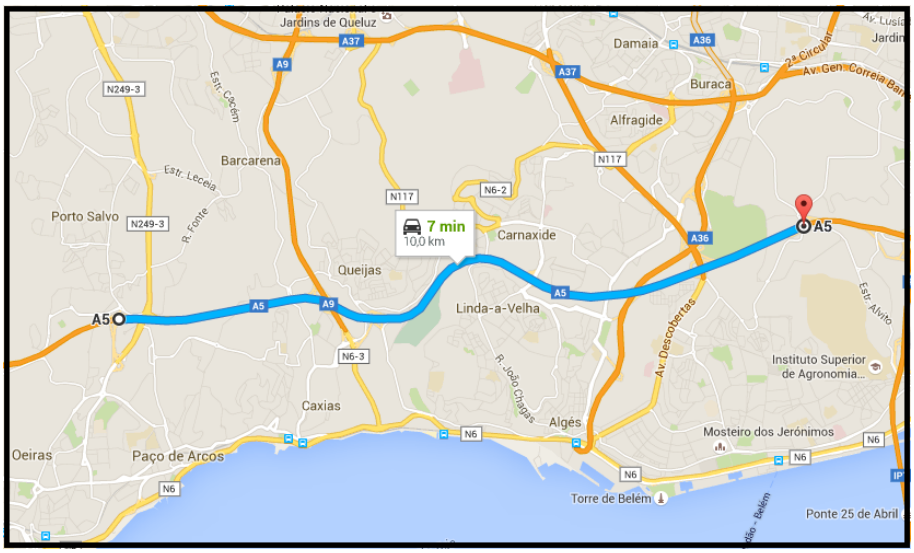


Figure 5.1: Portuguese A5 Highway (Lisboa-Cascais) with selected road section

It is important to note that in the test scenarios where congestion is reported, either in scenario II or scenario III, the decrease of the instantaneous velocity leading to congestion depends directly on the distance between road sensors, that is, we consider that the greater the distance between road sensors, the greater the number of vehicles (each vehicle considered to have 5m) necessary for the highway to be totally congested. Thus, the test cases present a more pronounced decrease of instantaneous velocity for shorter distances between road sensors than for larger distances. With that said, we consider that for each scenario where congestion occurs, there are three points of initial congestion subject to study, as illustrated in Fig. 5.2:

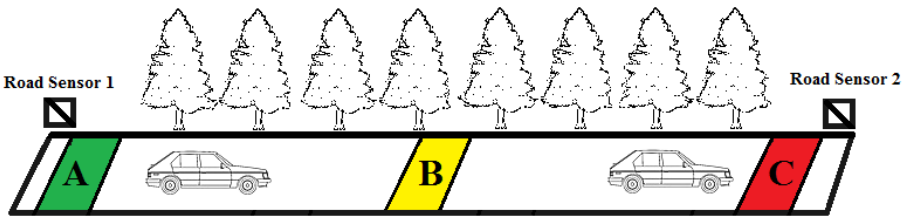


Figure 5.2: Three test cases. A - best case , B - intermediate case , C - worst case

- **A - best case**, congestion reported exactly after a road sensor.
- **B - intermediate case**, congestion reported in the middle of two road sensors.
- **C - worst case**, congestion reported exactly before a road sensor.

To conclude, it is necessary to state the parameterization values required by the simple threshold method, described in chapter 3, and used in **RoadAhead**, in particular its static parameters, which vary depending upon the type of road on which the system operates, namely: $V_{average}(1)$, β and γ. Since the test cases are simulated on a highway, these parameters were set with the following values for each road sensor, $V_{average}(1) = V_{instantaneous}(1)$, $\beta = 30km/h$ and $\gamma = 80km/h$. Thus, these values represent,

respectively, the first instantaneous velocity collected by a road sensor at the beginning of a test, and the lower and upper boundaries used to obtain an initial congestion state based on the maximum road velocity limit permitted by law in a Portuguese highway (120km/h) .

5.3 Scenario I - Free Flow

5.3.1 Simulation scenario

The first scenario (Free Flow) aims at evaluating the impact of the α parameter in the accuracy of the inferred traffic congestion under almost perfect road traffic conditions, that is, vehicles move with instantaneous velocities close to the maximum road velocity limit (100-120 km/h). Thus, in this scenario, it is expected a green color to be indicated as traffic congestion by **RoadAhead**. Therefore, in order to validate it, we developed several tests cases with various conditions, as depicted in Table 5.1, and with instantaneous velocities varying between 100 and 120 km/h. For the sake of simplicity, and as shown in Fig. 5.3, we selected test case with distance = 250m (between road sensor 20 and 21) to be used as example, being the rest of the test cases included in Appendix A.

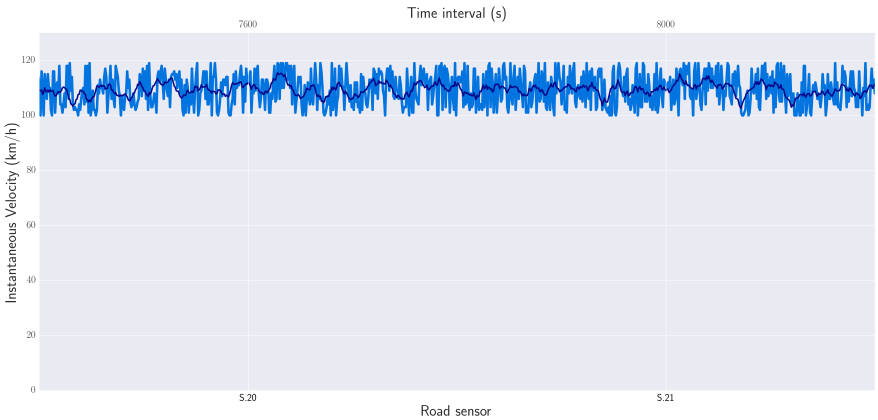


Figure 5.3: Free Flow - Input with distance = 250m

5.3.2 Results and discussion

Fig. 5.4, 5.5, 5.6 and 5.7 show the traffic congestion state inferred by **RoadAhead**, when using the test case illustrated in Fig. 5.3 with $\alpha = [0.25; 0.50; 0.75; 1]$. By analysing the figures, one can quickly notice that, due to the variation of the instantaneous velocities between 100 and 120 km/h, **RoadAhead** infers a green color of traffic congestion. In addition, by looking at the green lines, one can conclude that as we increase α we also increase the noise level in the system i.e. a higher α parameter discards older vehicle instantaneous velocities faster. Therefore, we concluded that a lower α, $\alpha = 0.25$, is a better value to be used by **RoadAhead**, since transient events can happen and change abruptly the corresponding traffic congestion color (i.e. $V_{average}(t) = V_{instantaneous}(t)$ with $\alpha = 1$, thus the test case illustrated in Fig. 5.3 is identical to the output represented in Fig. 5.7).

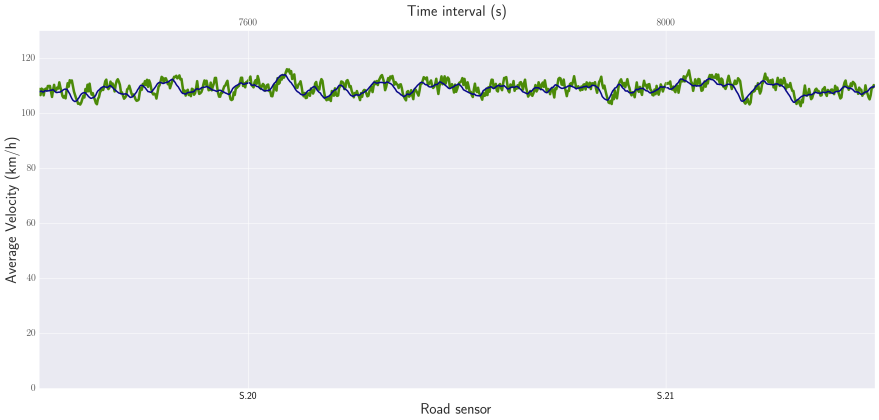


Figure 5.4: Free Flow - Output with distance = 250m and $\alpha = 0.25$

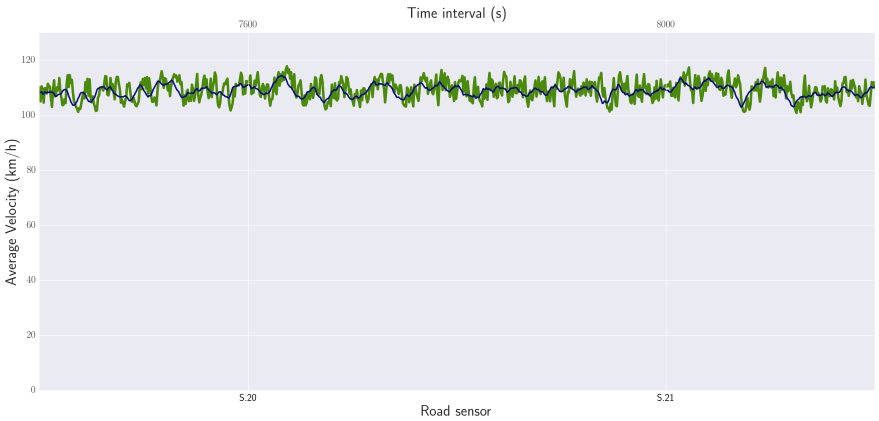


Figure 5.5: Free Flow - Output with distance = 250m and $\alpha = 0.50$

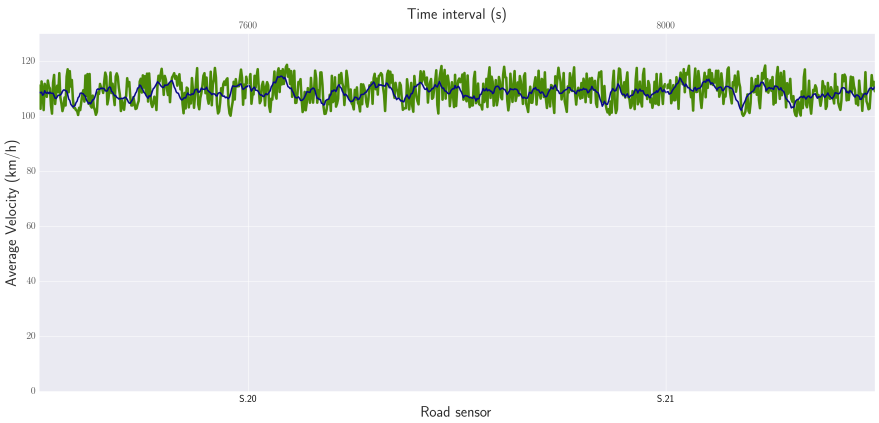


Figure 5.6: Free Flow - Output with distance = 250m and $\alpha = 0.75$

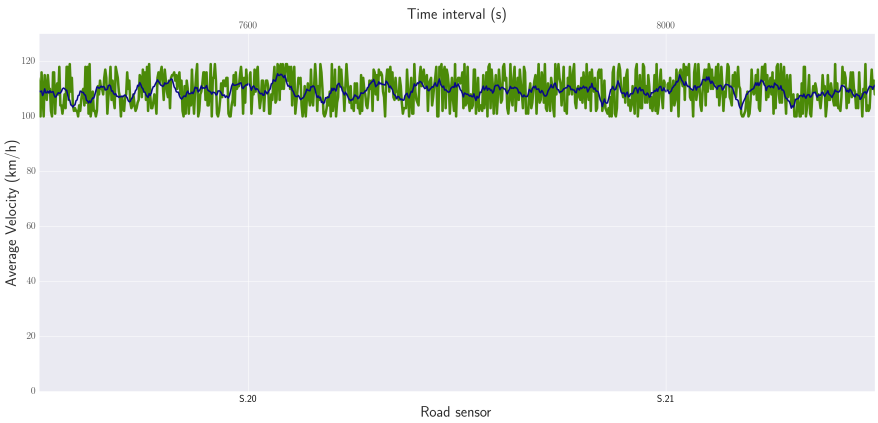


Figure 5.7: Free Flow - Output with distance = 250m and $\alpha = 1$

5.4 Scenario II - Slighty Congested

5.4.1 Simulation scenario

The second scenario (Slighty Congested) aims at evaluating the impact of the distance between road sensors in the accuracy and response-time of **RoadAhead**, when the road traffic is fluid (100-120 km/h) but starts to decrease until a point where it gets near the minimum road velocity limit (50 km/h). Thus, in this scenario, it is expected to be indicated by **RoadAhead** a green color followed by a yellow color as traffic congestion. Therefore, in order to validate it, we developed several test cases using the conditions depicted in Table 5.1, but with a particularity. As said in Section 5.2, we consider that for each scenario where congestion occurs, there are three points of initial congestion subject to study. Thus, these test cases were subject to the points A, B, and C. For the sake of simplicity, and as shown in Fig. 5.8, 5.9, and 5.10, we selected test cases A, B, and C, with distance = 250m (between road sensor 20 and 21) to be used as examples, being the rest of the test cases included in Appendix A.

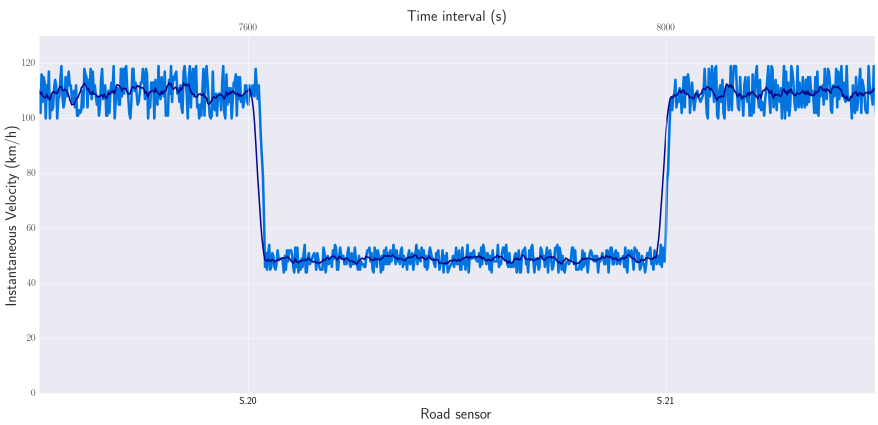


Figure 5.8: Slightly Congested - Case A - Input with distance = 250m

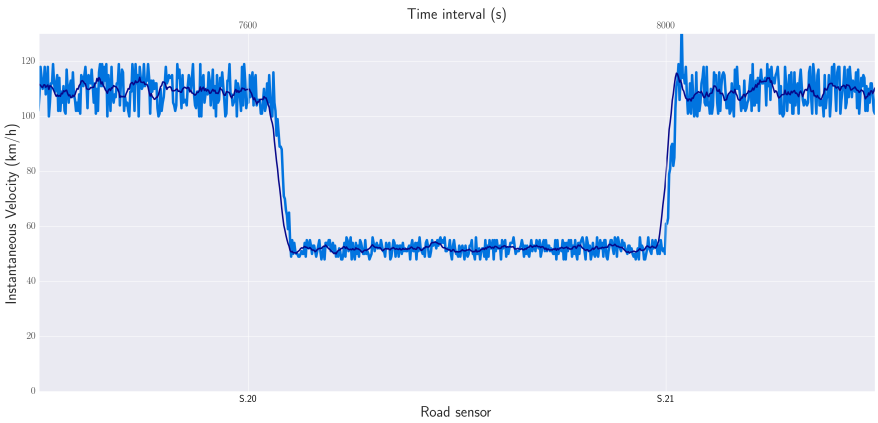


Figure 5.9: Slightly Congested - Case B - Input with distance = 250m

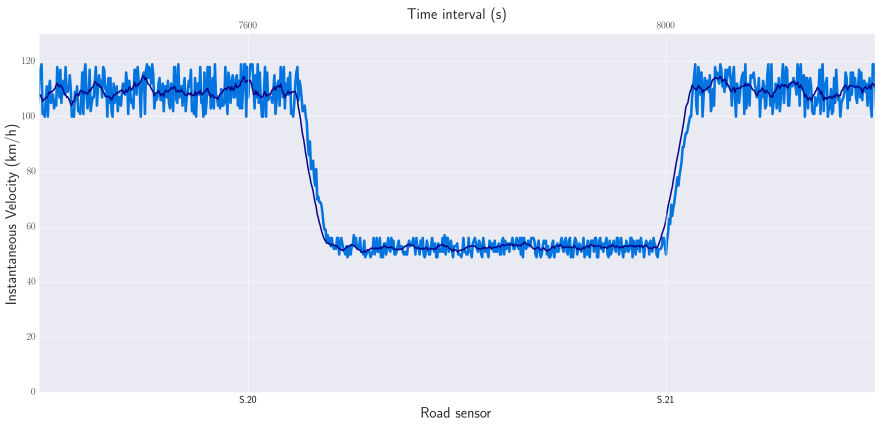


Figure 5.10: Slightly Congested - Case C - Input with distance = 250m

5.4.2 Results and discussion

Fig. 5.11, 5.12, and 5.13 show the traffic congestion state inferred by **RoadAhead**, when using respectively, the test cases illustrated in Fig. 5.8, 5.9, and 5.10, with $\alpha = 0.25$. By analysing the figures, one can initially see that **RoadAhead** detects a slightly traffic congestion between road sensor 20 and 21, due to a decrease of the instantaneous velocities until a point near the minimum road velocity limit (50 km/h). Therefore, we can observe that **RoadAhead** response-time is directly linked to the starting point of traffic congestion with a higher response-time, the farther away is the road sensor. This can be seen by measuring the elapsed time from the starting point of road sensor 20 until the detection of a transition between different traffic congestion states (from green to yellow). Additionally, this conclusion can also be taken if we increase the distance between road sensors, as illustraded in Fig. A.12, A.14, A.16, A.18, A.20, and A.22. With this said, we concluded that **RoadAhead** provides a better overview of the road network the lower the distances between road sensors.

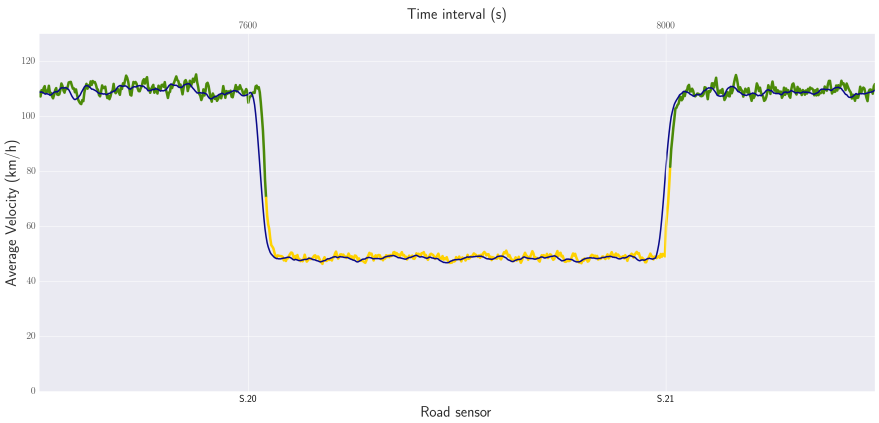


Figure 5.11: Slightly Congested - Case A - Output with distance = 250m and $\alpha = 0.25$

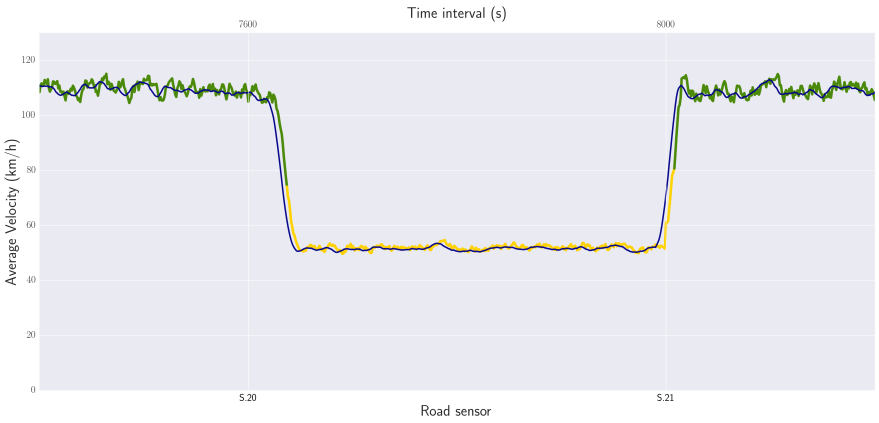


Figure 5.12: Slightly Congested - Case B - Output with distance = 250m and $\alpha = 0.25$

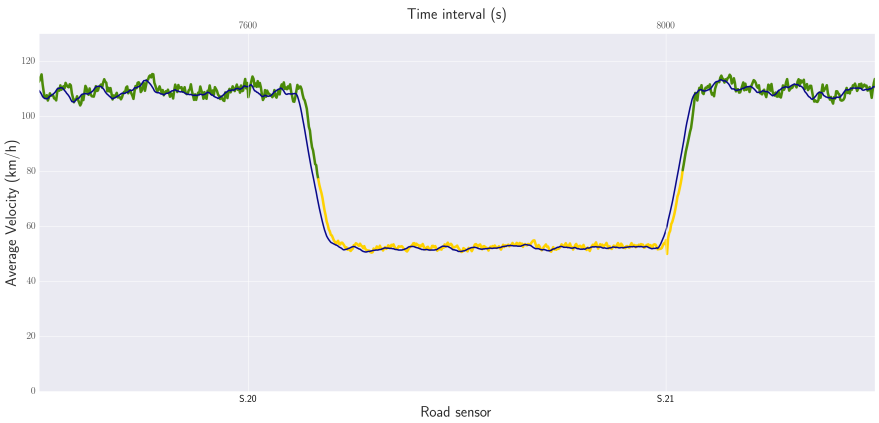


Figure 5.13: Slightly Congested - Case C - Output with distance = 250m and $\alpha = 0.25$

5.5 Scenario III - Emergency/Congested

5.5.1 Simulation scenario

Like the previous scenario, the final one (Emergency/Congested) also aims at evaluating the impact of the distance between road sensors in the accuracy and the response-time of **RoadAhead**. Although, in this case the road traffic is fluid (100-120 km/h) but it is forced to a complete stop due to a road accident. Thus, in this scenario, it is expected to be indicated by **RoadAhead** a green, followed by a very short yellow, and finally, a red color indicating a traffic jam. Therefore, in order to validate it, we consider the same test cases of the previous scenario (Slightly Congested), but with the particularity of points A, B, and C, being potential accident sites causing a drastic decrease of intantaneous velocities to around 10 km/h. For the sake of simplicity, and as shown in Fig. 5.14, 5.15, and 5.16, we selected test cases A, B, and C, with distance = 250m (between road sensor 20 and 21) to be used as examples, being the rest of the test cases included in Appendix A.

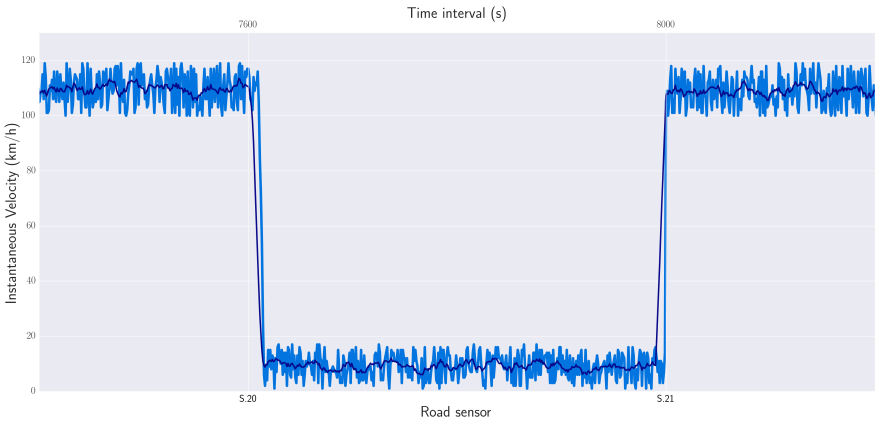


Figure 5.14: Emergency/Congested - Case A - Input with distance = 250m

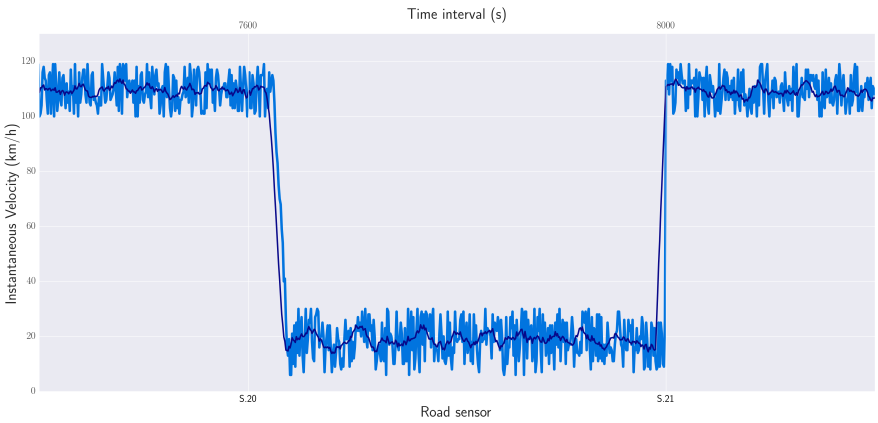


Figure 5.15: Emergency/Congested - Case B - Input with distance = 250m

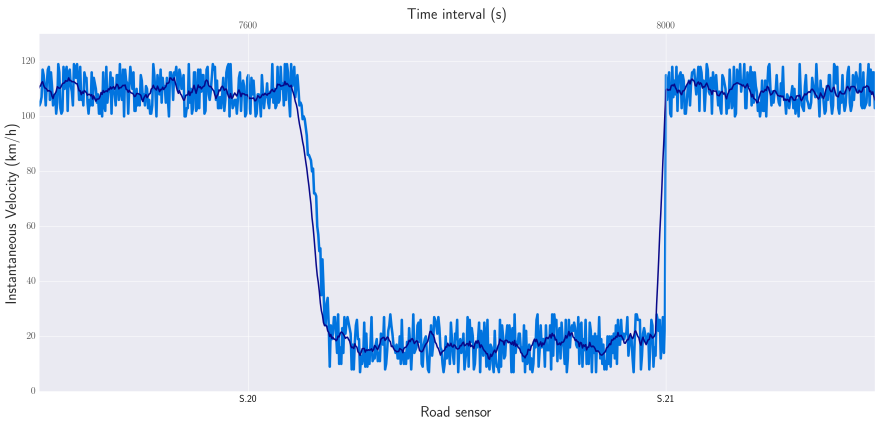


Figure 5.16: Emergency/Congested - Case C - Input with distance = 250m

5.5.2 Results and discussion

Fig. 5.17, 5.18, and 5.19 show the traffic congestion state inferred by **RoadAhead**, when using respectively, the test cases illustrated in Fig. 5.14, 5.15, and 5.16, with $\alpha = 0.25$. By analysing the figures, we concluded that the results are similar to the previous scenario (Slightly Congested), but with the particularity that **RoadAhead** is able to detect a heavy traffic congestion (after road sensor 20). Additionally, this conclusion can also be taken if we increase the distance between road sensors, as illustraded in Fig. A.24, A.26, A.28, A.30, A.32, and A.34.

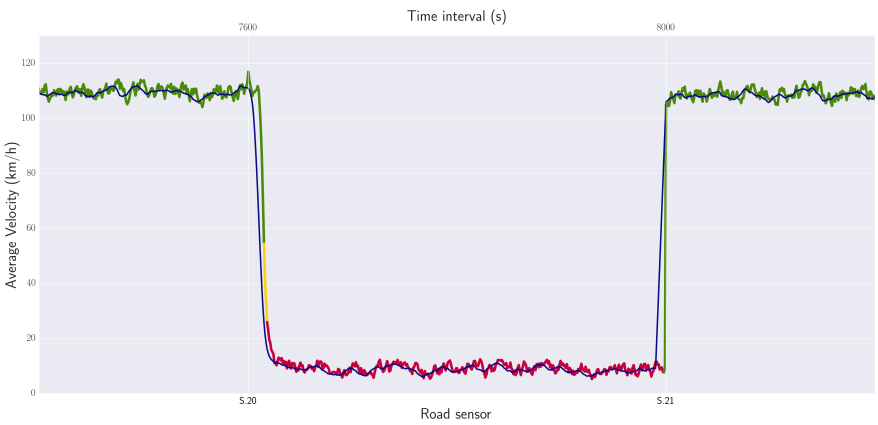


Figure 5.17: Emergency/Congested - Case A - Output with distance = 250m and $\alpha = 0.25$

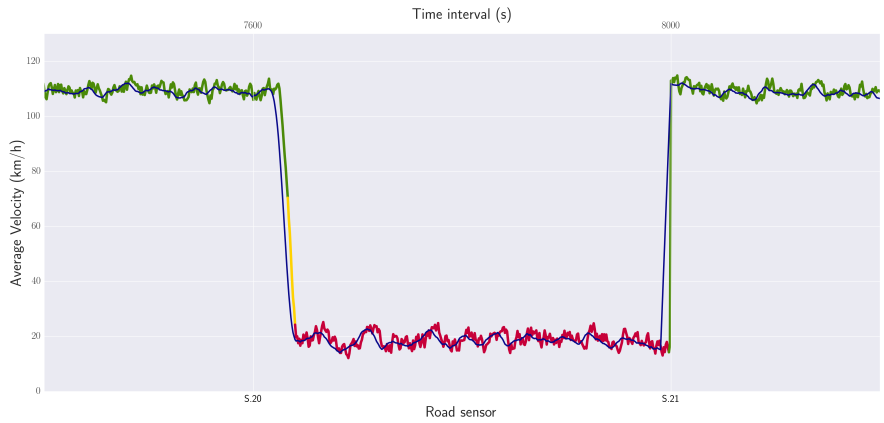


Figure 5.18: Emergency/Congested - Case B - Output with distance = 250m and $\alpha = 0.25$

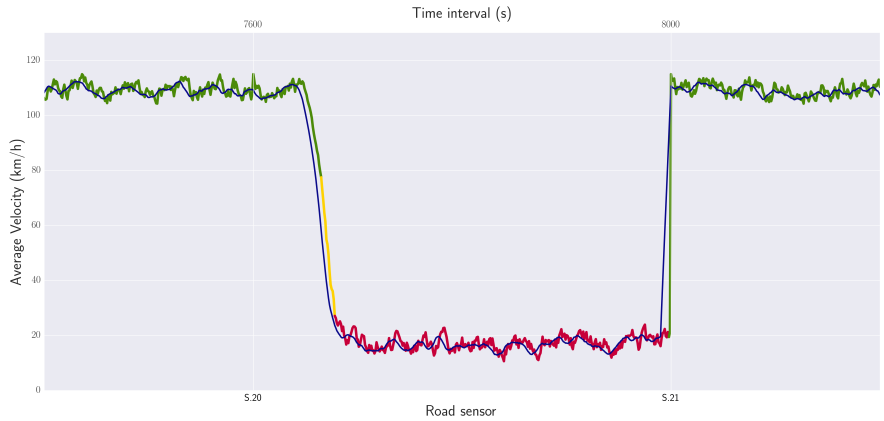


Figure 5.19: Emergency/Congested - Case C - Output with distance = 250m and $\alpha = 0.25$

5.6 Synthesis

In this chapter we evaluated **RoadAhead** by using the simulation tool described in chapter 4. Thus, we started by stating our simulation goals that aim at optimizing the conditions in which **RoadAhead** should operate, respectively, its static parameters (α, $V_{average}(1)$, β, and γ) and the distance between road sensors.

Furthermore, a set of scenarios were defined based on the proposed goals and setup, namely: **Free Flow**, when the road traffic is fluid and moving with velocity close to the maximum road velocity limit, **Slighty Congested**, when the road traffic is fluid but it starts to decrease until it gets near the minimum road velocity limit, and **Emergency/Congested**, when the road traffic is fluid but it is forced to move very slowly for an extended period of time or to a complete stop due to an road accident.

Afterwards, test cases were applied to these scenarios in a environment that simulates the Portuguese A5 highway (Lisboa-Cascais), more specifically, a 10 km long road section. However, we considered only a flat, one lane, straight road, with no traffic lights nor intersections, and with direction Cascais-Lisboa.

Finally, we discussed the test results and observed that **RoadAhead** operates at better conditions when we combine a lower distance between road sensors with a value $\alpha = 0.25$.

Chapter 6

Conclusions

In modern society, vehicles are a part of people's life. As the number of vehicles increases, more the traffic situation becomes a serious and complicated issue that can lead to several problems such as traffic congestion (i.e. traffic demand exceeding the roadway capacity) [1]. Therefore, this thesis goal was to develop a system called **RoadAhead** which offers a broader view of the traffic conditions in real-time by detecting existing vehicles on the road, and by displaying visual warnings to the vehicle drivers.

RoadAhead implements a decentralized architecture based on the existing Vehicle-to-Infrastructure (V2I) architecture. Thus, we saw that **RoadAhead** divides the existing road network into several well defined zones with variable size (e.g. $Zone_1 ... Zone_n$), each containing one or more roads, depending upon the traffic flow of each road. In addition, we stated that each zone relies on three major components which together manage to deliver the current traffic situation in real-time, namely: road sensors, a local server and light emitting diodes (LEDs).

Later on, we focused on describing the most relevant details regarding the implementation of **RoadAhead**, namely, its main components: road sensors, local server and light emitting diodes (LEDs). Thus, we started by explaining that due to the prohibitive cost of deploying a large sensor and light emitting diode network we followed the conventional approach used by most researches that rely on simulation tools. However, instead of using existing simulation tools we implemented our own due to the following reasons: **Flexibility** and **Complexity**.

Afterwards, we evaluated **RoadAhead** by using the simulation tool previously stated. Thus, we started by describing our simulation goals which aim at optimizing the conditions in which **RoadAhead** should operate, respectively, its static parameters (α, $V_{average}(1)$, β, and γ) and the distance between road sensors. Furthermore, a set of scenarios were defined based on the proposed goals and setup, namely: **Free Flow**, when the road traffic is fluid and moving with velocity close to the maximum road velocity limit, **Slighty Congested**, when the road traffic is fluid but it starts to decrease until it gets near the minimum road velocity limit, and **Emergency**/**Congested**, when the road traffic is fluid but it is forced to move very slowly for an extended period of time or to a complete stop due to an road accident. Next, we created several test cases which were applied to these scenarios in a environment that simulates

the Portuguese A5 highway (Lisboa-Cascais), more specifically, a 10 km long road section.

To conclude, we discussed the test results and observed that **RoadAhead** infers a better traffic congestion state whent we combine a lower distance between road sensors with a value $\alpha = 0.25$.

6.1 Future Work

As future work, we intend to validate the conclusions made in this document with a real-world experimental setup due to the fact that the evaluation of **RoadAhead** was done in a simulation environment with scenarios in ideal conditions and with manually generated instantaneous velocities. To do so, we would develop a small prototype of **RoadAhead** applied to the same highway but with a shorter distance due to the reasons presented in chapter 2 (e.g. high cost of road sensors).

Bibliography

[1] M. Al Jaafari, M. Al Shamisi, M. Al Darmki, M. Al Kaabi, A. Lakas, and M. Boulmalf, "Vie: A simulator for road traffic and inter-vehicular communication," in *Innovations in Information Technology, 2008. IIT 2008. International Conference on*, dec. 2008, pp. 548 –552.

[2] R. Hearne and A. Siddiqui, "Issues of dedicated lanes for an automated highway," in *Intelligent Transportation System, 1997. ITSC '97., IEEE Conference on*, nov 1997, pp. 619 –624.

[3] "Traffic congestion and reliability:-linking solutions to problems," *Cambridge Systematics, Inc and Texas Transportation Institute*, july 2004.

[4] S. Hollborn, "Intelligent transport system in japan," 2002.

[5] P. Masters, J. Lam, and K. Wong, "Incident detection algorithms for compass - an advanced traffic management system," in *Vehicle Navigation and Information Systems Conference, 1991*, vol. 2, 1991, pp. 295–310.

[6] L. Wischoff, A. Ebner, H. Rohling, M. Lott, and R. Halfmann, "Sotis - a self-organizing traffic information system," in *Vehicular Technology Conference, 2003. VTC 2003-Spring. The 57th IEEE Semiannual*, vol. 4, april 2003, pp. 2442 – 2446 vol.4.

[7] "The european communications architecture for co-operative systems: A key enabler for the development and wide scale deployment of co-operative systems for safe and environmentally sustainable transport in europe," *European Commission*, april 2009.

[8] T. Kosch, I. Kulp, M. Bechler, M. Strassberger, B. Weyl, and R. Lasowski, "Communication architecture for cooperative systems in europe," *Communications Magazine, IEEE*, vol. 47, no. 5, pp. 116 –125, may 2009.

[9] K. Dar, M. Bakhouya, J. Gaber, M. Wack, and P. Lorenz, "Wireless communication technologies for its applications [topics in automotive networking]," *Communications Magazine, IEEE*, vol. 48, no. 5, pp. 156 –162, may 2010.

[10] Y. C. Yang, C. M. Cheng, P. Y. Lin, and S. L. Tsao, "A real-time road traffic information system based on a peer-to-peer approach," in *Computers and Communications, 2008. ISCC 2008. IEEE Symposium on*, july 2008, pp. 513 –518.

[11] P. Davies, “The radio system-traffic channel,” in *Vehicle Navigation and Information Systems Conference, 1989. Conference Record*, sept. 1989, pp. A44 –A48.

[12] J.-W. Ding, C.-F. Wang, F.-H. Meng, and T.-Y. Wu, “Real-time vehicle route guidance using vehicle-to-vehicle communication,” *Communications, IET*, vol. 4, no. 7, pp. 870 –883, 30 2010.

[13] C. de Fabritiis, R. Ragona, and G. Valenti, “Traffic estimation and prediction based on real time floating car data,” in *Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on*, oct. 2008, pp. 197 –203.

[14] M. Wright, D. Stallings, and D. Dunn, “The effectiveness of global positioning system electronic navigation,” in *SoutheastCon, 2003. Proceedings. IEEE*, april 2003, pp. 62 – 67.

[15] T. Taleb, E. Sakhaee, A. Jamalipour, K. Hashimoto, N. Kato, and Y. Nemoto, “A stable routing protocol to support its services in vanet networks,” *Vehicular Technology, IEEE Transactions on*, vol. 56, no. 6, pp. 3337 –3347, nov. 2007.

[16] J. Miller, “Vehicle-to-vehicle-to-infrastructure (v2v2i) intelligent transportation system architecture,” in *Intelligent Vehicles Symposium, 2008 IEEE*, june 2008, pp. 715 –720.

[17] L. E. Y. Mimbela and L. A. Klein, “A summary of vehicle detection and surveillance technologies used in intelligent transportation systems,” *The Vehicle Detector Clearinghouse*, 2000.

[18] M. Minea, “Emc and related problems in detection of road and rail vehicles,” in *Telecommunications in Modern Satellite, Cable and Broadcasting Services, 2007. TELSIKS 2007. 8th International Conference on*, sept. 2007, pp. 581 –584.

[19] A. K. Lawrence, K. M. Milton, and R. P. G. David, *Traffic detector handbook: third edition - volume I*, ”third” ed. Federal Highway Administration, U.S. Department of Transportation, 2006.

[20] M. Dalgleish, “Vehicle detection for advanced transport telematics,” in *Road Traffic Monitoring and Control, 1994., Seventh International Conference on*, apr 1994, pp. 164 –167.

[21] C. Leung and W.-D. Hao, “Piezo electric sensors for monitoring airport surface movement - a sustainable airport ground traffic management system,” in *Integrated and Sustainable Transportation System (FISTS), 2011 IEEE Forum on*, 29 2011-july 1 2011, pp. 339 –343.

[22] “Evaluation of non-intrusive technologies for traffic detection,” *Minnesota Department of Transportation Office of Traffic Engineering/ITS Section and SRF Consulting Group, Inc*, september 2010.

[23] G. Challita, S. Mousset, F. Nashashibi, and A. Bensrhair, “An application of v2v communications : Cooperation of vehicles for a better car tracking using gps and vision systems,” in *Vehicular Networking Conference (VNC), 2009 IEEE*, oct. 2009, pp. 1 –6.

[24] L. Vanajakshi, “Centre of excellence in urban transport iit madras synthesis report on its including issues and challenges in india,” *dept. of Civil Engineering, IIT Madras*, december 2010.

[25] E. Kaplan, “Understanding gps : principles and characteristics,” *Artech House*, 1996.

[26] B. Hoh, M. Gruteser, H. Xiong, and A. Alrabady, “Enhancing security and privacy in traffic-monitoring systems,” *Pervasive Computing, IEEE*, vol. 5, no. 4, pp. 38 –46, oct.-dec. 2006.

[27] M. Schmidt, L. Giorgi, M. Chevreuil, S. Paulin, S. Turvey, and M. Hartmann, “Galileo: Impacts on road transport,” *JRC-IPTS Technical Report EUR 21865*, 2005.

[28] J. Lopes, J. Bento, E. Huang, C. Antoniou, and M. Ben-Akiva, “Traffic and mobility data collection for real-time applications,” in *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, sept. 2010, pp. 216 –223.

[29] B. Krause, C. von Altrock, and M. Pozybill, “Intelligent highway by fuzzy logic: congestion detection and traffic control on multi-lane roads with variable road signs,” in *Fuzzy Systems, 1996., Proceedings of the Fifth IEEE International Conference on*, vol. 3, sep 1996, pp. 1832 –1837 vol.3.

[30] S. Medley and M. Demetsky, “Development of congestion performance measures using its information,” *Virginia Transportation Research Council (VTRC)*, january 2003.

[31] L. Ying, “The metropolitan transportation performance evaluation [d],” *Beijing: Beijing Industry University*, 2002.

[32] D. Schrank, B. Eisele, and T. Lomax, “Urban mobility report,” *Texas transportation institute*, 2012.

[33] S. T. T. Lomax and G. Shunk, “Quantifying congestion,” *National Cooperative Highways Research Program (NCHRP), Transportation Institute College, Tech. Rep. 398*, 2001.

[34] M. Robitaille and T. Nguyen, “Road congestion assessment: from theory to practice: case of the greater montreal road network,” in *Transportation Association of Canada, Annual Convention*, 2003.

[35] W. Pattara-atikom, P. Pongpaibool, and S. Thajchayapong, “Estimating road traffic congestion using vehicle velocity,” in *ITS Telecommunications Proceedings, 2006 6th International Conference on*, June, pp. 1001–1004.

[36] F. Porikli and X. Li, “Traffic congestion estimation using hmm models without vehicle tracking,” in *Intelligent Vehicles Symposium, 2004 IEEE*, June, pp. 188–193.

[37] N.N, “fuzzytech 4.1 1 user’s and reference manual,” *INFORM Software*, 1996.

[38] P. Pongpaibool, P. Tangamchit, and K. Noodwong, “Evaluation of road traffic congestion using fuzzy techniques,” in *TENCON 2007 - 2007 IEEE Region 10 Conference*, 30 2007-Nov. 2, pp. 1–4.

[39] R. Qi-liang, “Research on h-fuzzy evaluation method for traffic congestion of city road-network,” in *Information Management, Innovation Management and Industrial Engineering, 2008. ICIII ’08. International Conference on*, vol. 3, Dec., pp. 41–44.

[40] A. Paplinsk, “Adaptive neuro-fuzzy inference system (anfis),” 20 May 2005.

[41] J. Lu and L. Cao, “Congestion evaluation from traffic flow information based on fuzzy logic,” in *Intelligent Transportation Systems, 2003. Proceedings. 2003 IEEE*, vol. 1, Oct., pp. 50–53 vol.1.

[42] C. Deng, F. Wang, H. Shi, and G. Tan, “Real-time freeway traffic state estimation based on cluster analysis and multiclass support vector machine,” in *Intelligent Systems and Applications, 2009. ISA 2009. International Workshop on*, May, pp. 1–4.

[43] J. MacQueen, “Some methods of classification and analysis of multivariate observations,” in *Proc. Fifth Berkeley Symp. on Math. Statist. and Prob. (Univ. of Calif. Press)*, vol. 1, 1967.

[44] M. B.-A. I. Kaysi and H. Koutsopoulos, “An integrated approach to vehicle routing and congestion prediction for real-time driver guidance,” in *Transp. Res. Rec. 1408, Transportation Research Board, Washington, D.C.*, 1993, pp. 66–74.

[45] A. Roshandeh and M. Nesheli, “Analysis of distance headway and density at a variable message signs network in kuala lumpur,” in *Cybernetic Intelligent Systems (CIS), 2010 IEEE 9th International Conference on*, 2010, pp. 1–4.

[46] A. Roshandeh and O. Puan, “Assessment of impact of variable message signs on traffic surveillance in kuala lumpur,” in *Intelligence and Security Informatics, 2009. ISI ’09. IEEE International Conference on*, 2009, pp. 223–225.

[47] R. D. Huchingson, C. L. Dudek, and W. Dorsey, “Highway advisory radio message development guide,” 1982.

Appendix A

Evaluation results

A.1 Scenario I - Free Flow

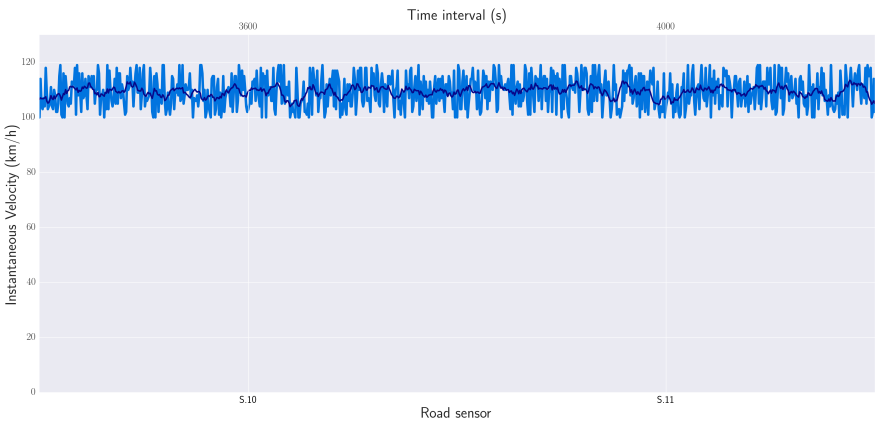


Figure A.1: Free Flow - Input with distance = 500m

Figure A.2: Free Flow - Output with distance = 500m and $\alpha = 0.25$

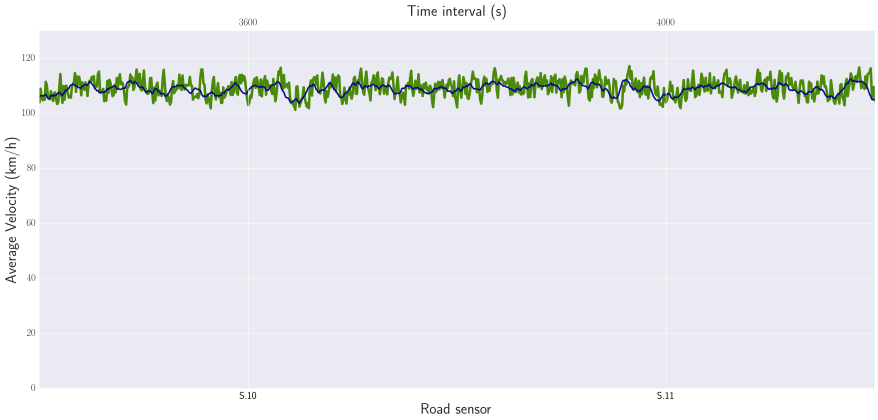


Figure A.3: Free Flow - Output with distance = 500m and $\alpha = 0.50$

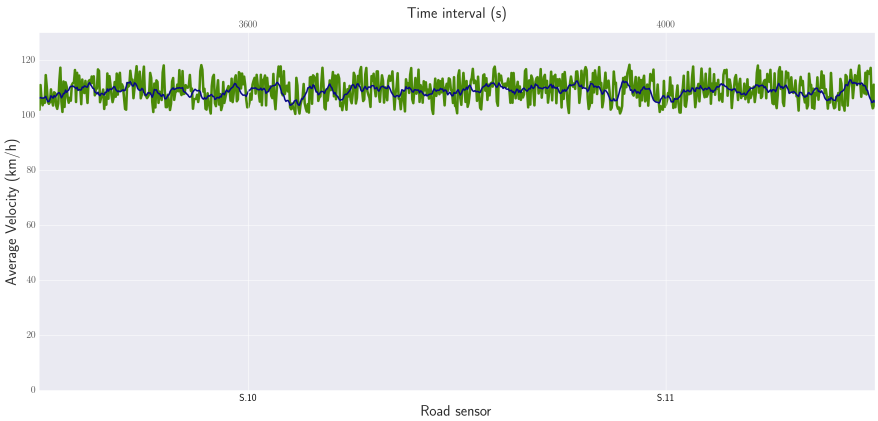


Figure A.4: Free Flow - Output with distance = 500m and $\alpha = 0.75$

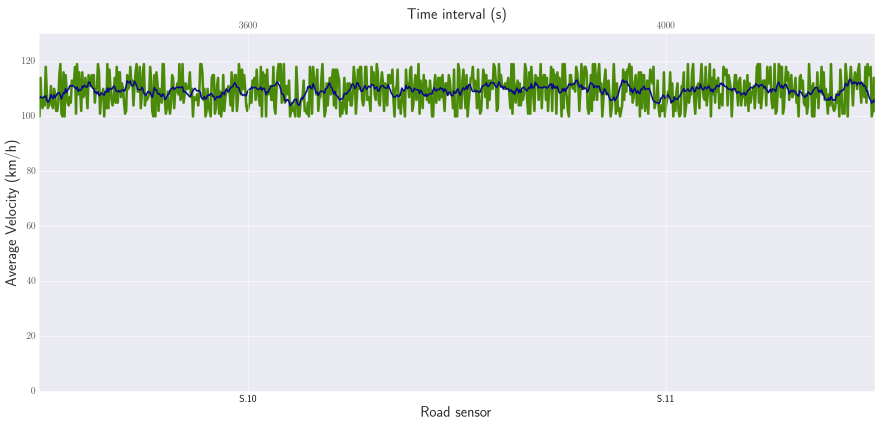


Figure A.5: Free Flow - Output with distance = 500m and $\alpha = 1$

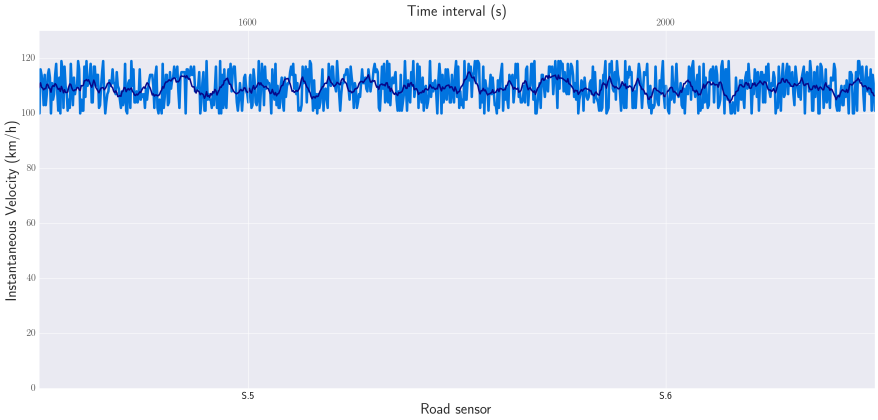


Figure A.6: Free Flow - Input with distance = 1000m

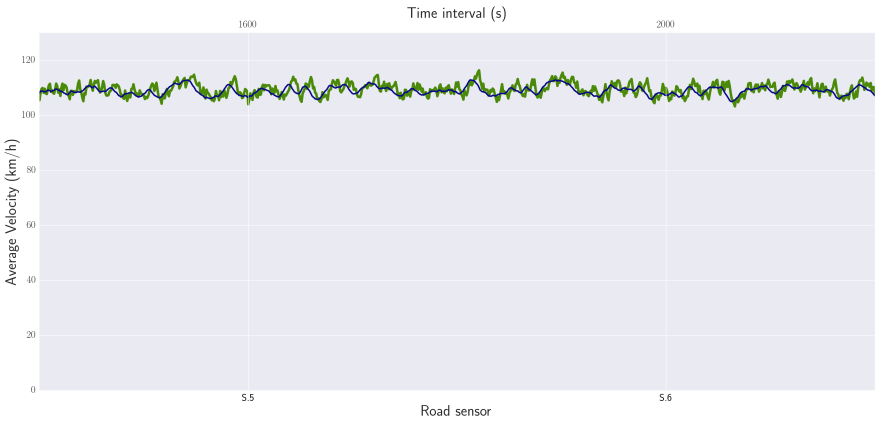


Figure A.7: Free Flow - Output with distance = 1000m and $\alpha = 0.25$

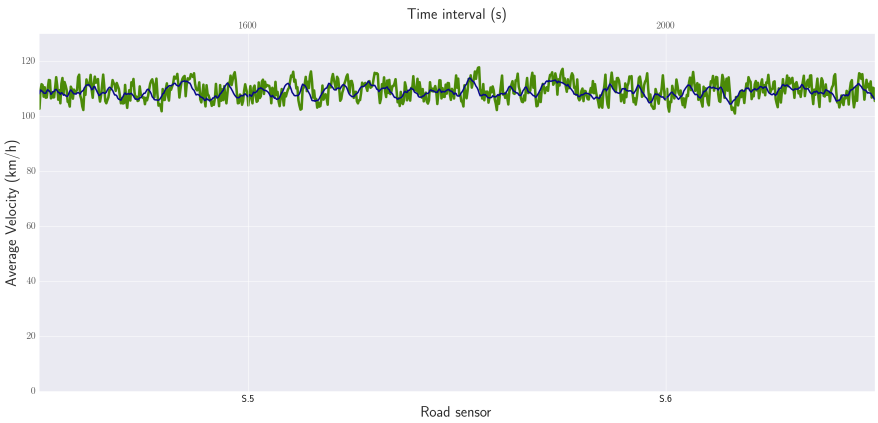


Figure A.8: Free Flow - Output with distance = 1000m and $\alpha = 0.50$

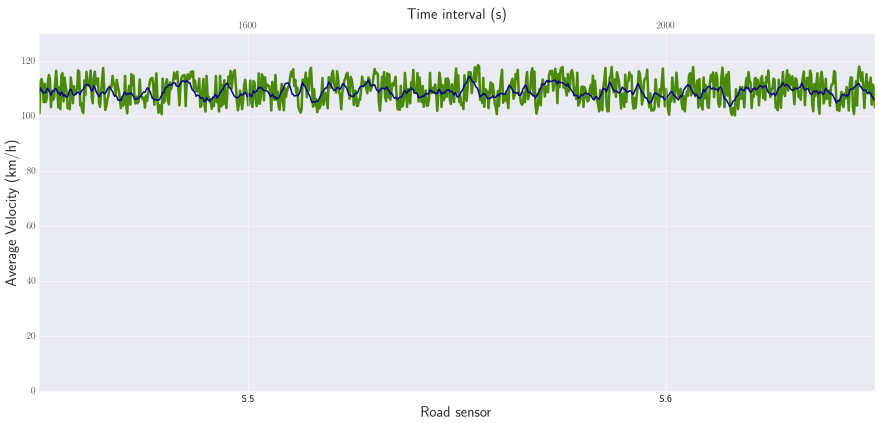


Figure A.9: Free Flow - Output with distance = 1000m and $\alpha = 0.75$

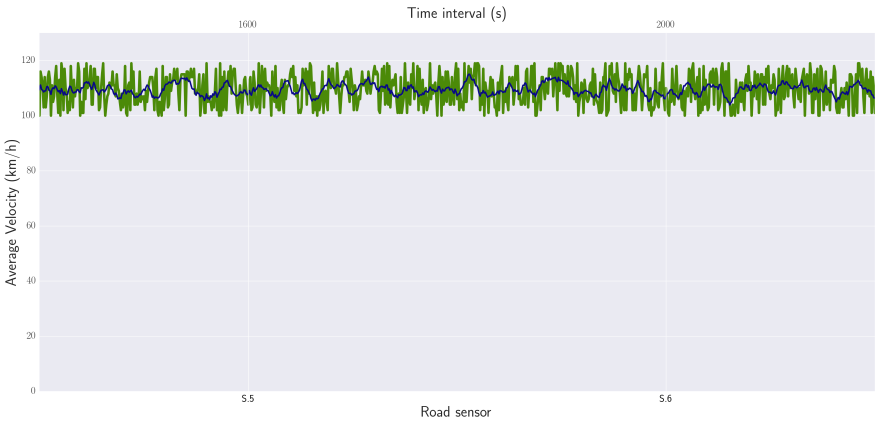


Figure A.10: Free Flow - Output with distance = 1000m and $\alpha = 1$

A.2 Scenario II - Slightly Congested

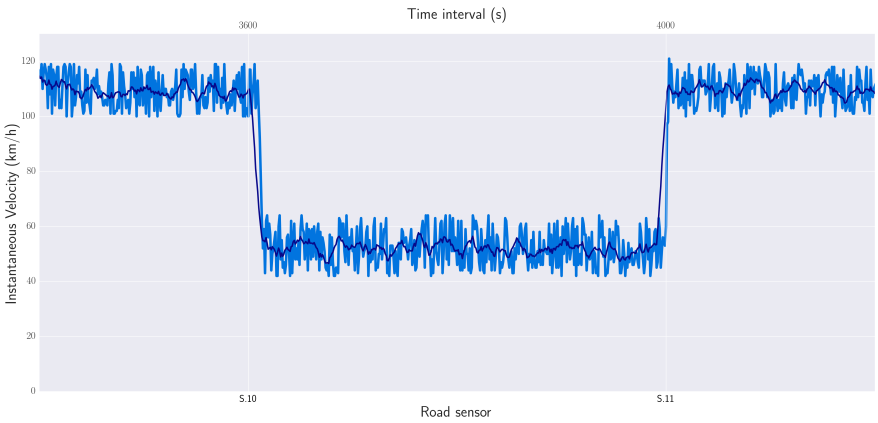


Figure A.11: Slightly Congested - Case A - Input with distance = 500m

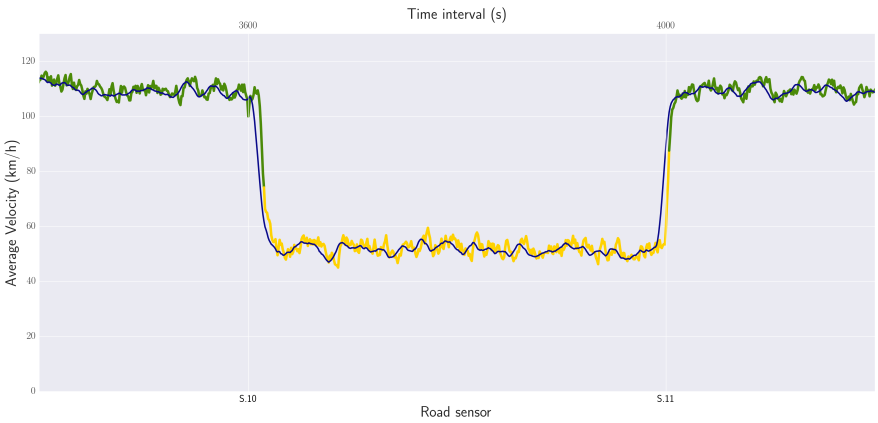


Figure A.12: Slightly Congested - Case A - Output with distance = 500m and $\alpha = 0.25$

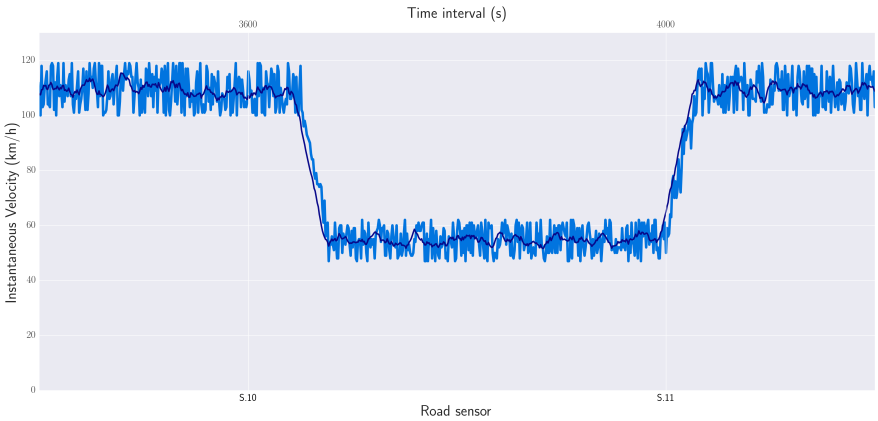


Figure A.13: Slightly Congested - Case B - Input with distance = 500m

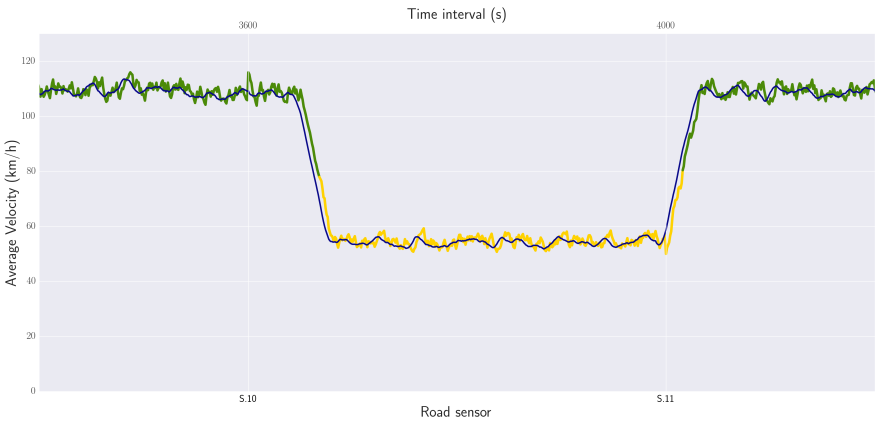


Figure A.14: Slightly Congested - Case B - Output with distance = 500m and $\alpha = 0.25$

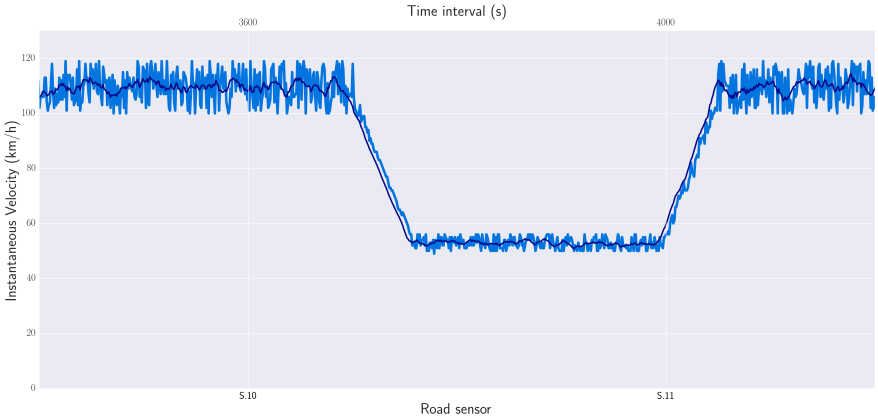


Figure A.15: Slightly Congested - Case C - Input with distance = 500m

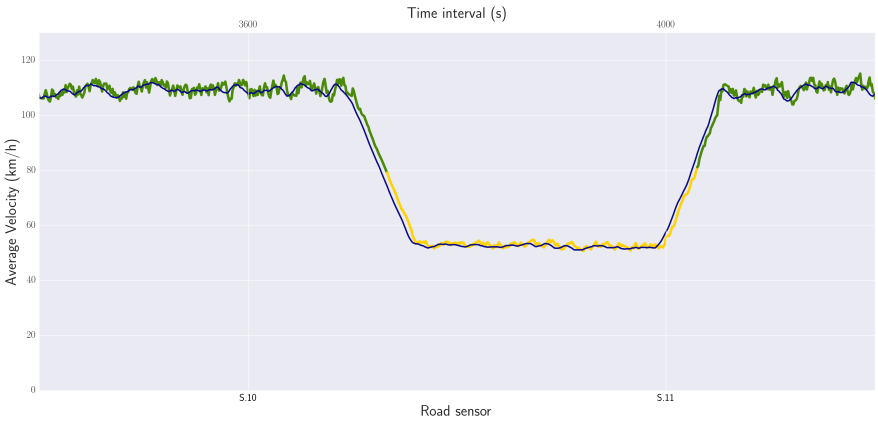


Figure A.16: Slightly Congested - Case C - Output with distance = 500m and $\alpha = 0.25$

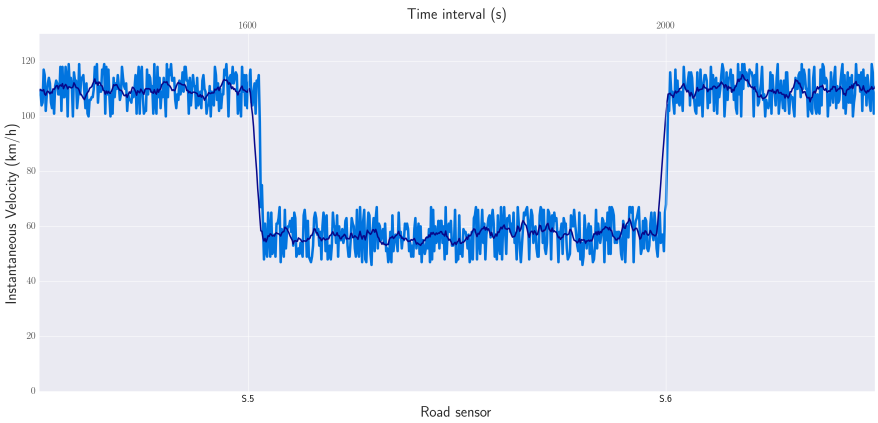


Figure A.17: Slightly Congested - Case A - Input with distance = 1000m

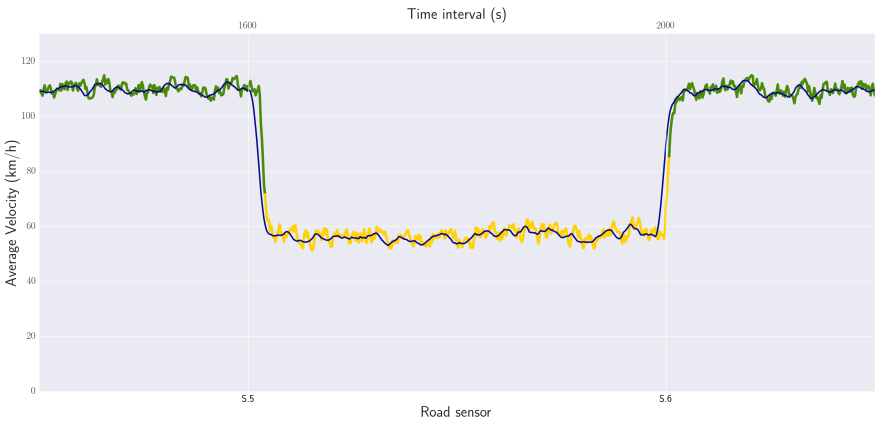


Figure A.18: Slightly Congested - Case A - Output with distance = 1000m and $\alpha = 0.25$

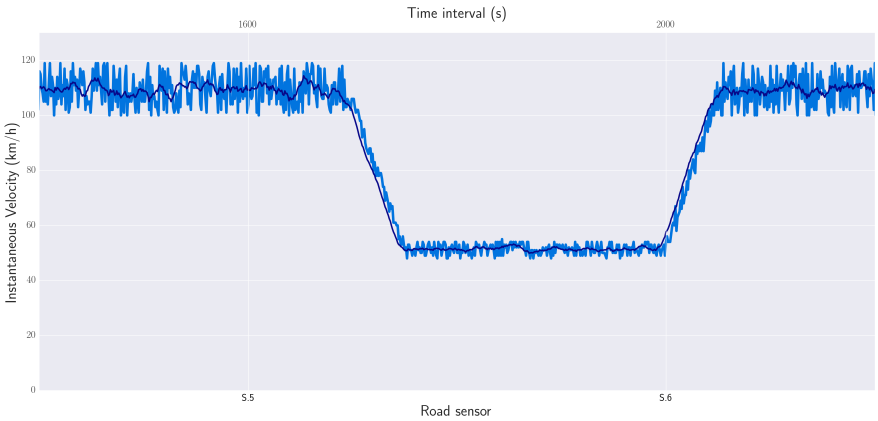


Figure A.19: Slightly Congested - Case B - Input with distance = 1000m

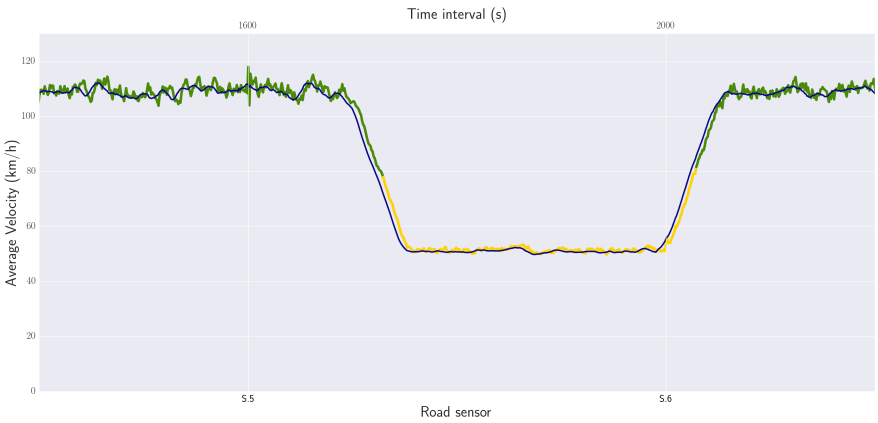


Figure A.20: Slightly Congested - Case B - Output with distance = 1000m and $\alpha = 0.25$

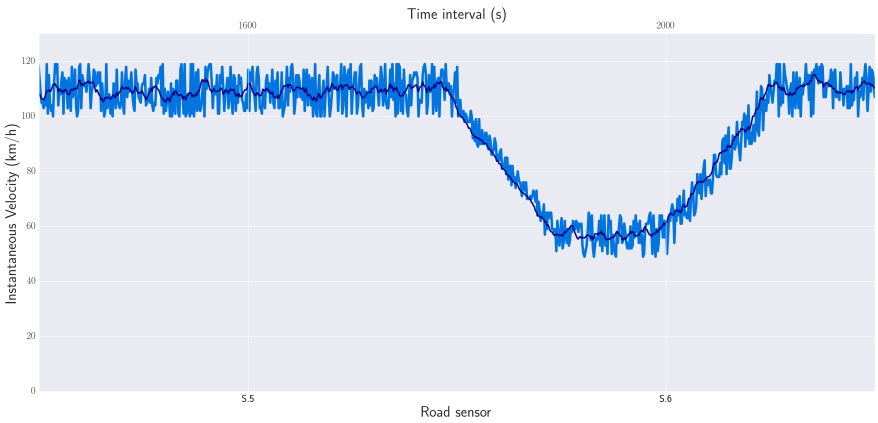


Figure A.21: Slightly Congested - Case C - Input with distance = 1000m

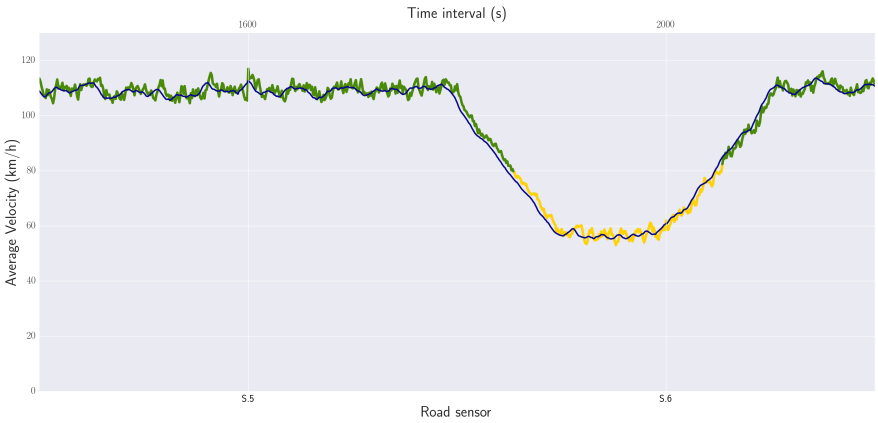


Figure A.22: Slightly Congested - Case C - Output with distance = 1000m and $\alpha = 0.25$

A.3 Scenario III - Emergency/Congested

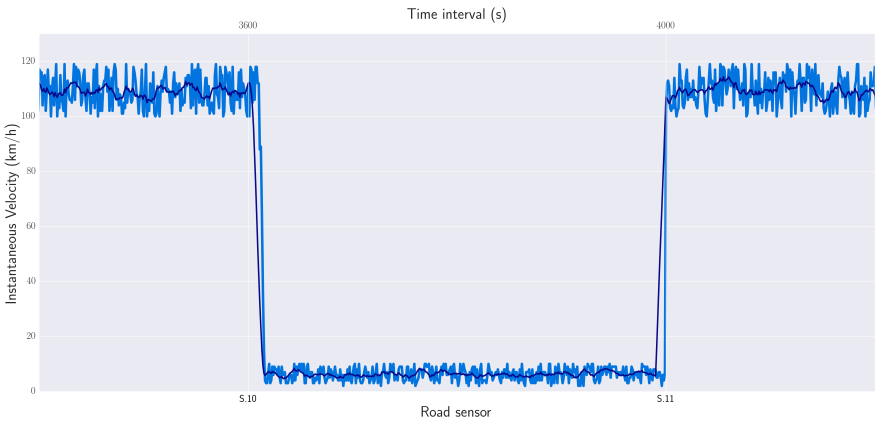


Figure A.23: Emergency/Congested - Case A - Input with distance = 500m

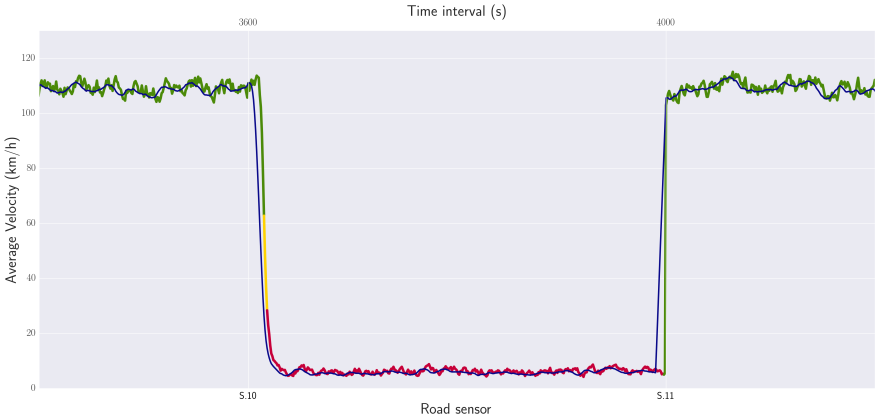


Figure A.24: Emergency/Congested - Case A - Output with distance = 500m and $\alpha = 0.25$

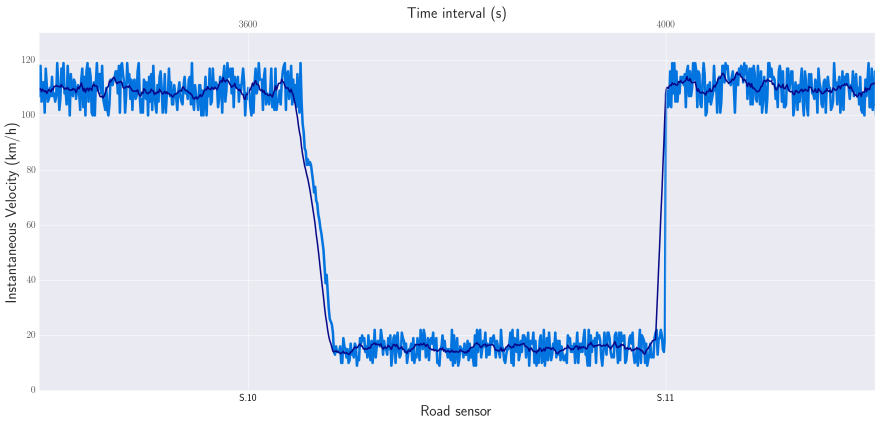


Figure A.25: Emergency/Congested - Case B - Input with distance = 500m

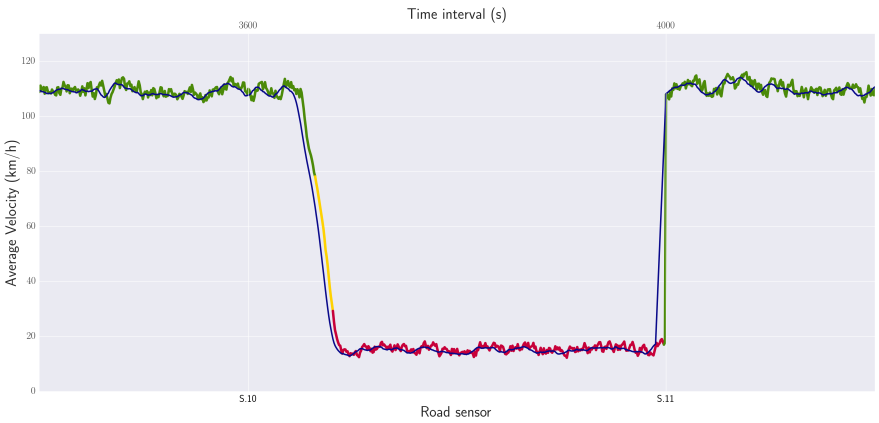


Figure A.26: Emergency/Congested - Case B - Output with distance = 500m and $\alpha = 0.25$

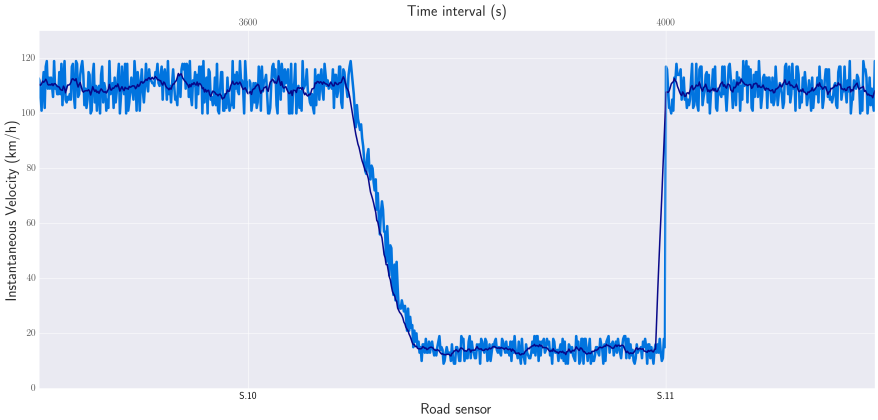


Figure A.27: Emergency/Congested - Case C - Input with distance = 500m

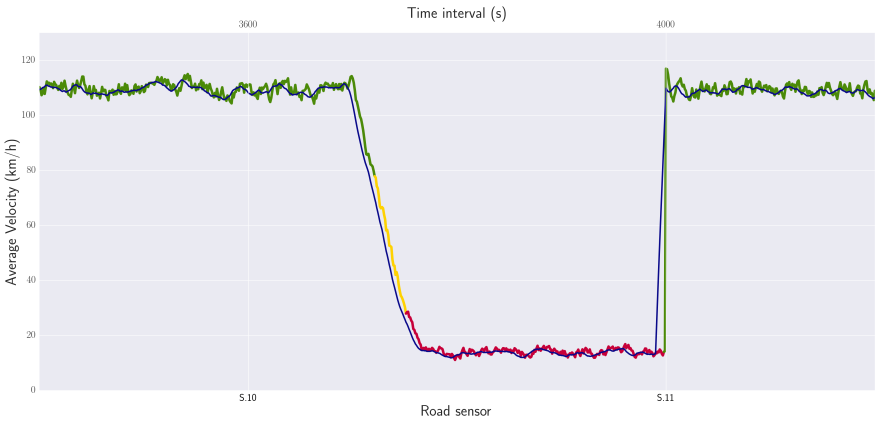


Figure A.28: Emergency/Congested - Case C - Output with distance = 500m and $\alpha = 0.25$

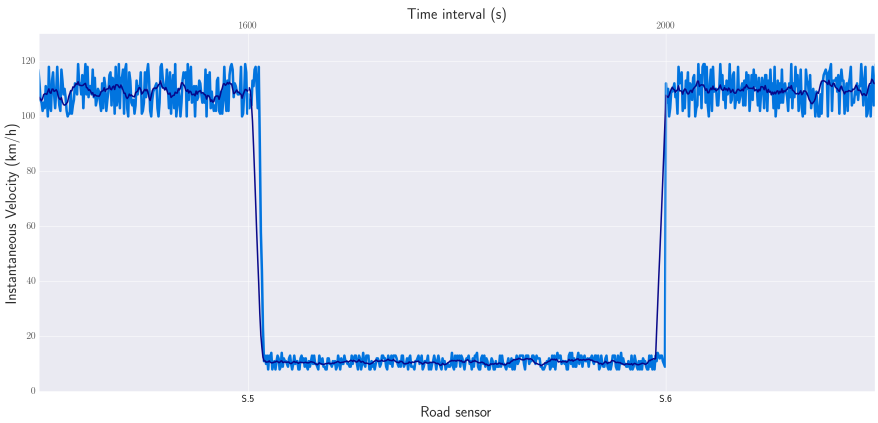


Figure A.29: Emergency/Congested - Case A - Input with distance = 1000m

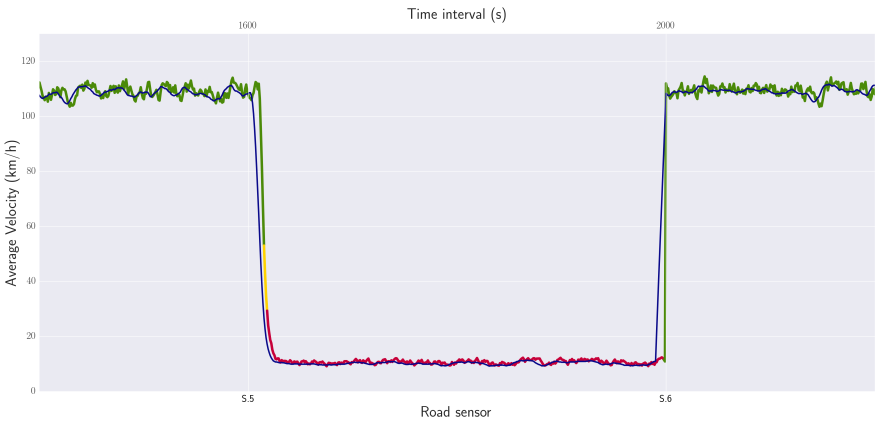


Figure A.30: Emergency/Congested - Case A - Output with distance = 1000m and $\alpha = 0.25$

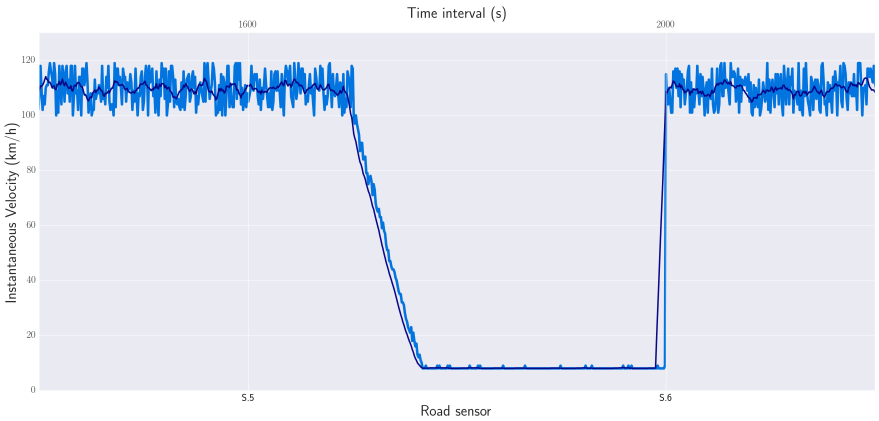


Figure A.31: Emergency/Congested - Case B - Input with distance = 1000m

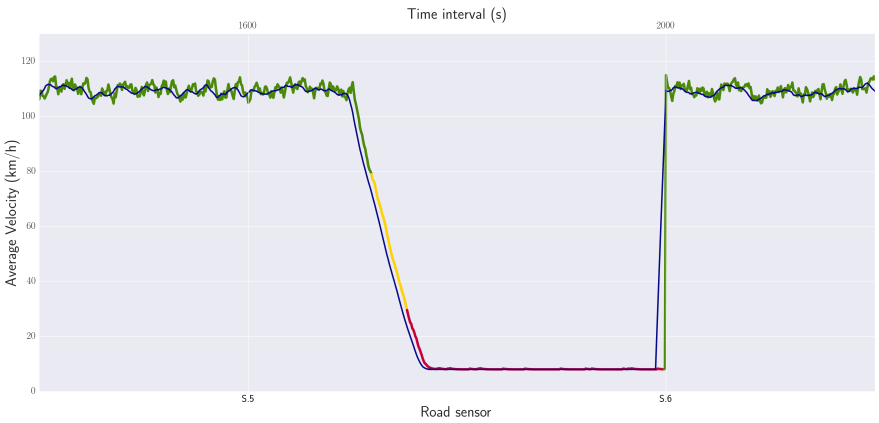


Figure A.32: Emergency/Congested - Case B - Output with distance = 1000m and $\alpha = 0.25$

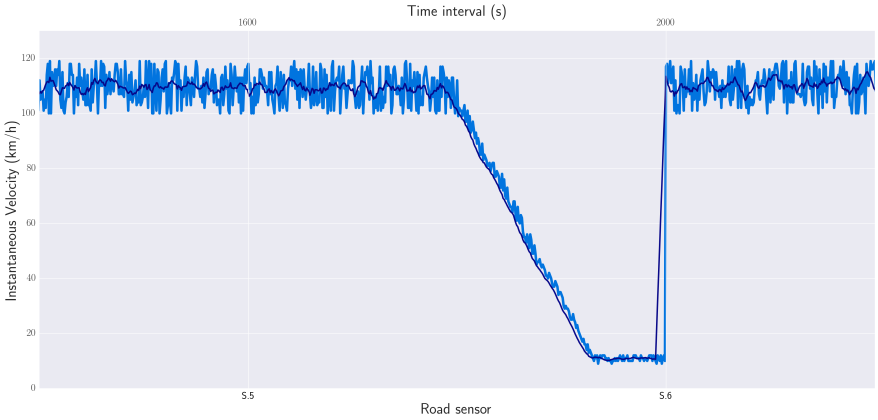


Figure A.33: Emergency/Congested - Case C - Input with distance = 1000m

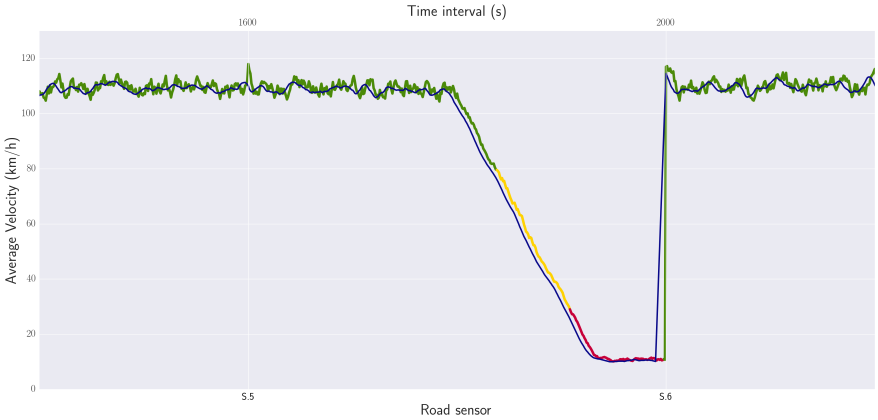


Figure A.34: Emergency/Congested - Case C - Output with distance = 1000m and $\alpha = 0.25$