Abstract

Intelligent Transportation Systems (ITS) can substantially improve road safety and traffic efficiency. This is possible by allowing communication among nearby vehicles and among vehicles and fixed roadside units. A popular standard for vehicular communications is IEEE 802.11p. It is based on a CSMA/CA MAC method that does not guarantee channel access in a finite time and so is not suitable for real-time communications. It also needs methods to control and limit the load, since the transmission of periodic information among vehicles can saturate the channel. In this thesis, a new real-time scheduling algorithm suitable for ITS applications is introduced. It is based on a TDMA MAC method, where the roadside unit has the tasks to estimate the channel conditions and assign fractions of time slot to users. A linear programming approach is considered to minimize an index of utility of the transmissions. Multi-hop communication scenarios among the vehicles are considered for both uplink and downlink communications. It is shown how the optimal duration of the fraction of time slot depends on the channel conditions. A higher channel gain corresponds to a higher transmission time whereas a lower channel gain corresponds to a lower transmission time. It is concluded that the approach studied in the thesis can guarantee a high utility provided that the complexity of the optimization is reduced as the number of involved vehicles increases.
Contents

Abstract

1 Introduction

1.1 Problem formulation .......................................................... 2

2 Intelligent Transportation Systems ........................................... 3

2.1 Classification of Intelligent Transportation Systems ................. 3

2.2 Examples of devices for ITS ................................................. 11

3 Vehicular ad-hoc networks .................................................... 18

3.1 VANET architecture .......................................................... 22

3.2 Classification of the applications .......................................... 24

3.3 Examples of VANET applications ........................................... 26

3.4 Standards for VANET ......................................................... 27

3.4.1 Background on IEEE 802.11 ............................................. 27

3.4.2 IEEE 802.11p ................................................................. 30

3.4.3 IEEE 802.16e ................................................................. 36

3.4.4 ETSI ITS-G5 ................................................................. 36

4 Mathematical background ..................................................... 38

4.1 Linear programming .......................................................... 38

4.1.1 Duality ........................................................................... 40

4.1.2 Graphical method .......................................................... 41

4.1.3 The simplex method ....................................................... 42

4.1.4 General problem ........................................................... 43

5 Scheduling algorithm ............................................................. 45

5.1 Scheduling algorithm for real-time services .............................. 45

5.2 Proposed scheduling algorithm ............................................. 46

5.2.1 Uplink .......................................................................... 47

5.2.2 Downlink ........................................................................ 63
6 Simulation results
   6.1 Scenario 1 ................................. 80
   6.2 Scenario 2 ................................. 84
   6.3 Scenario 3 ................................. 88

7 Conclusions and future works 92

Bibliography 93
# List of Figures

2.1 Advanced traffic management systems ........................................... 4 
2.2 GPS ...................................................................................... 5 
2.3 Changeable message sign .......................................................... 6 
2.4 Driver monitoring system ......................................................... 6 
2.5 Collision avoidance system ....................................................... 7 
2.6 Advanced public transportation system ...................................... 7 
2.7 Commercial vehicle operations ............................................... 8 
2.8 Electronic payment system ...................................................... 9 
2.9 Automatic vehicle identification ............................................. 9 
2.10 Emergency management systems .......................................... 10 
2.11 Vulnerable individual protection systems ............................... 10 
2.12 Radar system mounted on Audi A8 ........................................ 12 
2.13 Example of radar system that applies the brakes ....................... 13 
2.14 Possible urban scenario ........................................................ 13 
2.15 Disturbance produced by the vehicle in earth’s magnetic field 14 
2.16 VDS240 .............................................................................. 14 
2.17 Implementation of VDS240 .................................................... 15 
2.18 SNAPS server ....................................................................... 15 
2.19 WSN for vehicle classification ............................................... 16 
2.20 Example of WSN setup .......................................................... 16 
2.21 Packaging of the sensors ........................................................ 17 
2.22 Example of deployment of vibration sensors ........................... 17 
3.1 Operation of a cognitive system placed inside a vehicle ............... 19 
3.2 Architecture of the proposed functionality ............................... 20 
3.3 Functionality components and indicative information flow .......... 21 
3.4 Example of a VANET architecture ........................................ 22 
3.5 DSRC channel assignment in America .................................... 23 
3.6 Channel allocation in Europe ................................................... 23 
3.7 Intersection violation warning ............................................... 28 
3.8 Electronic brake warning ........................................................ 29 
3.9 On-coming traffic warning ....................................................... 30
2006 studies [1] estimated about 1.25 millions dead and 30 millions injured each year due to car accidents. These shocking data stimulated the development of new technologies to make streets safer, using interdisciplinary competences in telecommunications, informatics and electronics. Progress has been made thanks to wireless communications, mobile computing and sensing capabilities. Systems that improve street safety through these technologies are commonly referred to as Intelligent Transportation Systems (ITS).

Although air, water and rail transportation systems use advanced systems, ITS only concern road transportation systems, where vehicles interact with the environment, among them and intelligent infrastructure. Wireless technology is needed to ensure communication among these entities. Allowing these communications permits the exchange of useful information among vehicles like speed, size and density of vehicles on the streets.

ITS have not only the purpose to improve safety, but also to improve transport efficiency, avoiding possible traffic congestions, that leads to less fuel consumption and less travel time. The reduction of fuel consumption is also important to environmental protection, as pollutant emission would be reduced. To improve transport efficiency is possible to build new streets, but this is a deleterious approach both for its high price and the negative influence on environment. A fairer approach would be to distribute sensing systems along the streets that collect information from moving vehicles. These can provide to drivers guidance on alternative roads in order to avoid congestions.

ITS do not give benefits only to drivers but also to governmental authorities. In fact, with the assistance of these technologies is possible to easily control the streets and verify the respect of the speed limits, but also change roadsigns through changeable signs directly controlled by traffic control center.

In this context, vehicular ad-hoc networks (VANETs) play an important
role for ITS because they permit the exchange of useful information among nearby vehicles and among vehicles and fixed roadside units.

1.1 Problem formulation

The most important feature in VANETs is the scheduling policy, and so the optimal allocation of the resources among multiple vehicles. In the vehicular environment, the scheduling policy has to face the two following issues:

- Emergency messages are classified as real-time;
- The transmission of periodic information among vehicles can saturate the channel.

In this thesis we want to solve these two problems by introducing a new real-time scheduling algorithm, that avoids the congestion of the queues of the vehicles, suitable for ITS applications.

The thesis is organized as follows:

- Chapter 2: we give a classification of Intelligent Transportation Systems and some examples of devices already in use;
- Chapter 3: we introduce VANET architecture and the standards for vehicular communications, focusing on IEEE 802.11p;
- Chapter 4: we introduce linear programming;
- Chapter 5: we describe the proposed scheduling algorithm;
- Chapter 6: we give the results obtained by Matlab simulations;
- Chapter 7: we draw the conclusions and we give some recommendations for future works.
Chapter 2

Intelligent Transportation Systems

In this chapter we give a classification of ITS based on functionalities and required technologies and then we propose some examples of devices for ITS.

2.1 Classification of Intelligent Transportation Systems

ITS can be classified according to service users. This partition would contain many groups, such as car drivers, truck drivers, emergency vehicles or public transport drivers. Since many groups could use the same service, the classification based on customers is not convenient: it is preferable to make a classification based on functionalities and required technologies.

As proposed in [2], it is possible to identify nine ITS subsystems that can work together or independently, depending on the application. These subsystems are the following:

1. Advanced Traffic Management Systems;
2. Advanced Traveler Information Systems;
3. Advanced Vehicle Control and Safety Systems;
4. Advanced Public Transportation Systems;
5. Commercial Vehicle Operations;
6. Electronic Payment Systems;
7. Emergency Management Systems;
8. Vulnerable Individual Protection Systems;

In the following we describe these subsystems:

**Advanced Traffic Management Systems**

Advanced Traffic Management Systems (ATMS) collect information on the roads and forward them to a traffic control center, which analyzes them and determines control strategies for system optimization. At the end of the process it relays the results to every user. The technologies used in this case are detecting, sensing, communications and control. Possible applications are automatic incident detection, automatic vehicle identification and electronic toll collection. An example of this subsystem is given in Figure 2.1, where it is possible to observe the interaction among control center, infrastructures and vehicles.

![Figure 2.1: Advanced traffic management systems](image-url)
Advanced Traveler Information Systems

Advanced Traveler Information Systems (ATIS) collect data from different sources and, after having processed them, make them available to users, who use them to better plan their travel. Possible technologies that provide information to users are television at home, radio in vehicles, Global Positioning Systems (GPS) (see Figure 2.2) and changeable message signs (see Figure 2.3).

![Figure 2.2: GPS (source: http://jimmystreets.com/cars/use-of-gps-in-cars-car-tracker/)](image)

Advanced Vehicle Control and Safety Systems

Advanced Vehicle Control and Safety Systems (AVCSS) are useful in reducing the number of accidents and the probability of human errors caused mostly by driver fatigue. It is based on the principle that, increasing automation level within the vehicle, safety increases too and the probability of accidents decreases. The involved technologies are sensors, onboard computers, communications, electronics and control. Examples of applications are driver monitoring systems (see Figure 2.4) and collision avoidance systems, that slow down the vehicle when it is too close to the preceding (see Figure 2.5).
Advanced Public Transportation Systems

Advanced Public Transportation Systems (APTS) apply to public transport (buses, trains) ATMS, ATIS and AVCSS technologies to improve the services quality. Users have the benefit of knowing in real-time position and estimation of the arrival time of the transports, while public transport agencies have
better control and management of the fleet. An example of this subsystem is given in Figure 2.6.
Commercial Vehicle Operations

Commercial Vehicle Operations (CVO) apply to commercial vehicles (trucks) ATMS, ATIS and AVCSS technologies. By means of GPS it is possible to know in real-time the vehicle position, which is sent to a control center. In the control center, a fleet management system processes the information about position and then addresses commercial vehicles in an efficient way. Transport companies use this technique to track goods once they are on the road (see Figure 2.7).

![Figure 2.7: Commercial vehicle operations](image)

Electronic Payment Systems

Electronic Payment Systems (EPS) permit drivers to pay toll without stopping the vehicle thanks to the communications among onboard units and roadside units (see Figure 2.8). In this subsystem it is also included automatic vehicle identification, that allows to determine the identity of a vehicle subject to toll (see Figure 2.9).

Emergency Management Systems

Emergency Management Systems (EMS) are used for emergency vehicles, such as ambulances, fire trucks and police cars that have to reach as quick as possible their destination. A control center determines the best route with less delay and changes roadsigns along the chosen route. An example of this subsystem is given in Figure 2.10.
Figure 2.8: Electronic payment system (source: http://www.mhi.co.jp/en/products/detail/electronic_toll_collection_system.html)

Figure 2.9: Automatic vehicle identification: when a vehicle passes under a laser detector, two pictures of the number plate are taken, one of the front and one of the rear (source: http://www.transportstyrelsen.se/en/road/Congestion-tax/Congestion-tax-in-stockholm/How-do-control-points-work/)

Vulnerable Individual Protection Systems

Vulnerable Individual Protection Systems (VIPS) are dedicated to people with disabilities, elder, pedestrians and cyclists, that are the most vulnerable categories on the streets. As shown in Figure 2.11, these systems can use audio signals to make blind people understand when they can cross the road.
Figure 2.10: Emergency management systems

Figure 2.11: Vulnerable individual protection systems
Information Management Systems

Information Management Systems (IMS) are the basis of ITS and their task is to collect and analyze all the useful data in order to provide the requested information to every user category. For example, these information are used by drivers to plan travels in order to increase safety and efficiency and to reduce fuel costs. Transport agencies use them to better manage fleet on the road and to reduce operating costs. Authorities use them to improve performance on the roads, choosing how to arrange or modify road signs.

2.2 Examples of devices for ITS

There are many ITS applications: for example, it is possible to pay toll without stopping the vehicle, to stop the vehicle when the traffic light is red, to control the respect of the speed limits and the absence of unauthorized vehicles on public transport lanes. To perform these tasks it is possible to use many devices, such as those based on video, where cameras are installed on poles. These technologies are expensive and, above all, not reliable in case of bad weather conditions (fog, snow, rain). Another device is radar, that can be installed on vehicles to anticipate collisions. An example is given in the article [6], which explains how the radar system mounted on Audi A8 works (see Figure 2.12). This radar system works across three stages:

1. When the vehicle gets too close to the preceding, the system emits an alarm sound and a dashboard warning light comes on;
2. If there has been no change in the vehicle movement, the system makes a sudden short braking to try to warn the driver;
3. If the course remains unchanged, the radar system takes control and applies the brakes trying to prevent or, at least, soften the bumping (see Figure 2.13).

Wireless sensors directly buried under the road are the most used devices in ITS. Main advantages are low cost, battery alimentation, low power consumption, radio communication and independence from weather conditions. Figure 2.14 shows a possible wireless sensors deployment to improve city traffic.

In [3] it is proposed the use of magnetic sensors to determine presence, direction and speed of the vehicles. Their functioning is related to a magnetic field of known direction and strength that is disturbed by a moving vehicle. Earth’s magnetic field is taken as reference magnetic field and anisotropic
magnetoresistive (AMR) sensor are used to measure it. Vehicles contain steel, therefore they have the capacity to concentrate the flux lines of the Earth’s magnetic field. Hence, their presence is detectable by observing the magnetic flux variation in a given point (see Figure 2.15).

In [4] it is proposed a local area wireless sensor network for traffic surveillance called VDS240. Vehicle detection is the basis of this system and all other measures depend on the reliability of this process. VDS240 consists of the following three elements (see Figure 2.16 and Figure 2.17):

- **VSN240**: They are the wireless magnetic sensors placed in the center of each lane to detect the passage of vehicles. They are installed in less than 10 minutes.

- **AP240**: It is the access point placed on the side of the road that communicates via radio with nodes on the ground. Its task is to extract
Figure 2.13: The black car approaches too quickly to the preceding and the radar system applies the brakes to prevent or soften the bumping (source: Stevenson 2011)

Figure 2.14: Possible urban scenario: circles denote sensors for vehicles detection whereas squares denote access points that communicate via radio with sensors (source: Haoui, Kavaler and Varaiya 2007)

useful information from data collected by nodes.

- RP240: It is a repeater that extends access point communication range.
A possible application of this system is to calculate the speed of a vehicle. To do this it is sufficient to place two nodes on the same lane at a given distance. The access point receives information from these two nodes and calculates the average speed as the distance between the two nodes divided by the difference between the arrival times of the vehicle on them. Users can also connect via internet to SNAPS server, which collects information from access points on a certain area, to see what happens on the nodes in real-time (see Figure 2.18).

In [5] a wireless sensor network (WSN) is proposed to classify vehicles in transit on a certain lane based on belonging class (cars, busses, trucks). Vehicles are assigned to the different classes according to the axle number and the distance between them. WSN consists of the following three elements (see Figure 2.19 and Figure 2.20):
• Vibration sensors (see Figure 2.21): They contain an accelerometer and measure vibrations of the road caused by a moving vehicle. They are used to detect and count axles and to calculate their spacing. Using only one of these sensors at the center of the lane is not sufficient because vehicles never travel on a perfect straight line. Therefore, more vibration sensors are installed to detect all the axles (see Figure 2.22).
• Vehicle detection sensors (see Figure 2.21): They are based on magnetometers and measure variations of the magnetic field in presence of a vehicle. They are used to detect time of arrival $t_a$ and time of departure $t_d$ of the vehicles. In cooperation with the access point it is possible to calculate speed and length of the vehicles placing two sensors $(i,j)$ on the same lane at a given distance $d_{ij}$. The speed of the vehicle is obtained through $v = d_{ij}/|t_{aj} - t_{ai}|$, where $t_{ai}$ and $t_{aj}$ are the times of arrival on $i$ and $j$. The length of the vehicle can be estimated as
\[ L = v(t_{dj} - t_{aj}). \]

- Access point: It collects data from every sensor.

Figure 2.21: Packaging of the sensors in a sealed case that protects electronics from rain, water, oil spills etc (source: Bajwa, Rajagopal, Varaiya and Kavaler 2011)

Figure 2.22: On the left it is shown a truck that does not travel in a straight line, whereas on the right it is shown the deployment of the vibration sensors (source: Bajwa, Rajagopal, Varaiya and Kavaler 2011)

In the next chapter we introduce vehicular ad-hoc networks, that are useful for ITS applications.
Chapter 3

Vehicular ad-hoc networks

Vehicular ad-hoc networks (VANETs) are a typology of mobile ad-hoc network that allow communication among nearby vehicles and among vehicles and fixed roadside units (RSUs). Vehicular networks use wireless technology and are spontaneously formed among moving vehicles equipped with wireless interfaces, using short-range to medium-range communication systems.

Modern vehicles are equipped with several sensors, through which they process and analyze several data about the vehicle itself. VANET are used to increase the number of available information, by exchanging useful data among vehicles via broadcast communications. Thus, important safety information are transmitted to all nearby vehicles.

In [7] it is proposed to use cognitive networking principles, developing systems with learning abilities in order to have the perception of potential dangers and to consequently modify vehicles behavior. Cognitive systems keep track of past decisions in order to create a basic knowledge for future decisions (see Figure 3.1). So the reliability of taken decisions increases, as well as safety and efficiency on the streets. Traffic can be evaluated through models obtained by learning process: traffic conditions can change suddenly or recurrently, so only real time evaluation is not enough. The architecture of the proposed management functionality is shown in Figure 3.2, where it is possible to notice that vehicles form a sensor network where they can communicate with each other. As we can see in Figure 3.2, it is possible to identify two domains, the vehicle and the infrastructure. Vehicle Cognitive Management Functionality (V-CMF) works according to the following model:

- Information about vehicle status (velocity, position, direction), road conditions, traffic lights and roadsigns are received.

- Commands are sent to drivers to modify the vehicle behavior on the road.
Figure 3.1: Operation of a cognitive system placed inside a vehicle (source: Dimitrakopoulos and Demestichas 2010)

- Acquired information are processed and interpreted to get a basic knowledge and experience for future decisions.

Infrastructure Cognitive Management Functionality (I-CMF) works according to the following model:

- Information about elements or segments of the transport infrastructure (traffic lights, road signs, road conditions) are received.

- Policies set by the authorities are received, too.

- The configuration of elements and segments of the transport infrastructure is set.

- Collected information are kept to accelerate and improve future decisions.

In Figure 3.3 it is shown the information exchange among the different components.

Technical issues about VANETs are the following:

- Unfavorable scenario for wireless communications due to the multiple reflecting objects that reduce strength and quality of the received signal.
Fading effects due to the mobility of the vehicles.

Lack of a communication coordinator, that does not permit to solve the hidden node problem. Two vehicles, that are not in the same cell, cannot hear each other, so they can send packets at the same time causing a packet collision on a receiver vehicle reachable by both of them. This leads to a non-efficient use of the channel and to a high number of collisions.

Involvement of many participants, unlike other ad-hoc networks that have limited dimension.

High mobility of the nodes, which lead to work in highly dynamic conditions, where it is possible to have high or low density and speed of vehicles. For example, speed reaches up to 180 km/h on the highways with a density of 1 or 2 vehicles within 1 km, whereas in the cities, during rush hour, it is possible to find a huge density of vehicles that can reach up to 60 km/h.
Vehicular networks are usually partitioned, since it is possible to find isolated clusters of nodes.

Network topology changes frequently as vehicles are moving and nodes connect and disconnect very often.

Need of quality of service to assign priority and distinguish the various types of communications.

Emergency messages require a reliable and fast dissemination and are classified as real-time. If they do not arrive within a certain deadline they are useless and in some cases they can also cause serious problems to traffic safety systems.

Beacons are the messages that are periodically transmitted in broadcast by each vehicle. They are the backbone of safety applications and provide identity and information about the sender vehicle, such as velocity, position and direction.
3.1 VANET architecture

In a VANET, every vehicle is provided with an onboard unit (OBU), that is the communication device that allows Vehicle-to-Vehicle (V2V) and Vehicle-to-Roadside (V2R) communications. The smartest way to deploy RSUs is in critical points, such as traffic lights, intersections or stop signs. Furthermore, vehicles can access the internet by means of RSUs. An example of a VANET architecture is shown in Figure 3.4.

![Figure 3.4: Example of a VANET architecture](image)

75 MHz of dedicated short-range communications (DSRC) spectrum at 5.9 GHz have been allocated in America for V2V and V2R communications. The main purpose is always to enable applications that improve safety and efficiency, but comfort applications are also possible to push people to adopt this technology. As shown in Figure 3.5, DSRC spectrum is divided in seven channels, that are 10 MHz wide. Channel 178 is the control channel and is dedicated to safety communications. The two channels at the edges are reserved for future applications, whereas the rest are service channels used for any sort of application.

Three bands have been defined in Europe as follows:

- ITS-G5A, from 5.875 to 5.905 GHz, dedicated to safety applications;
• ITS-G5B, from 5.855 to 5.875 GHz, dedicated to non-safety applications;

• ITS-G5C, from 5.470 to 5.725 GHz, that can be used for ITS applications but is a radio local area network (RLAN) band.

The channel spacing in ITS-G5A and ITS-G5B is 10 MHz, so five channels are available (see Figure 3.6). G5CC is the control channel and the rest are service channels.

Figure 3.5: DSRC channel assignment in America (source: Moustafa and Zhang, Eds., 2009)

Figure 3.6: Channel allocation in Europe (source: Ström 2011)
3.2 Classification of the applications

As proposed in [8], applications can be divided in the following three categories:

- Safety-oriented applications (see Table 3.1): they help the driver to avoid potential dangers via the exchange of information among vehicles. Collected information concern the status of nearby vehicles and road conditions. They are the most important applications because they serve to avoid accidents. They can take control of the vehicle in case of dangerous situations, as in the case of the automatic braking, or only send warning messages to drivers.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intersection violation warning</td>
<td>It warns drivers when they are going to pass over a red light.</td>
</tr>
<tr>
<td>Electronic brake warning</td>
<td>It reports to the driver that a preceding vehicle has performed a sudden braking.</td>
</tr>
<tr>
<td>On-coming traffic warning</td>
<td>It helps the driver during overtaking maneuvers.</td>
</tr>
<tr>
<td>Vehicle stability warning</td>
<td>It alerts drivers that they should activate the vehicle stability control system.</td>
</tr>
<tr>
<td>Lane change warning</td>
<td>It helps drivers to perform a safe lane change.</td>
</tr>
<tr>
<td>Traffic signal violation warning</td>
<td>A roadside unit sends messages in broadcast to warn drivers of potential violations of traffic signals.</td>
</tr>
<tr>
<td>Post-crash notification</td>
<td>A vehicle involved in an accident sends warning messages in broadcast to approaching vehicles.</td>
</tr>
</tbody>
</table>

Table 3.1: Examples of safety-oriented applications

- Convenience-oriented applications (see Table 3.2): they serve to improve the efficiency of the roads and to save drivers time and money.

- Commercial-oriented applications (see Table 3.3): they serve to make the travel more comfortable and productive, for example, by means of the internet connection.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intersection management</td>
<td>V2V and V2R communications allow a better intersections management.</td>
</tr>
<tr>
<td>Limited access and detour warning</td>
<td>A roadside unit sends information in broadcast about limited access areas or possible detours.</td>
</tr>
<tr>
<td>Electronic toll collect</td>
<td>A vehicle establishes unicast communication with a toll gate roadside unit and pays the toll without stopping.</td>
</tr>
<tr>
<td>Congested road notification</td>
<td>A vehicle in a congested road sends information in broadcast to other vehicles.</td>
</tr>
<tr>
<td>Parking availability notification</td>
<td>A vehicle asks to a roadside unit for a list of available parking spaces, and the roadside unit sends the list to the vehicle.</td>
</tr>
</tbody>
</table>

Table 3.2: Examples of convenience-oriented applications

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Remote diagnosis</td>
<td>The driver can start a wireless connection with the dealer in order to upload the vehicle diagnostics information to detect possible problems.</td>
</tr>
<tr>
<td>Media or map download</td>
<td>A vehicle can start a wireless connection with the home network or a hot-spot to download maps and multimedia contents.</td>
</tr>
<tr>
<td>Service announcement</td>
<td>Restaurants and other businesses can use a roadside unit to send promotional messages to the drivers of the vehicles that are in their communication range.</td>
</tr>
</tbody>
</table>

Table 3.3: Examples of commercial-oriented applications
3.3 Examples of VANET applications

As presented in [9], there are five possible safety applications in VANETs:

- Intersection violation warning
- Electronic brake warning
- On-coming traffic warning
- Vehicle stability warning
- Lane change warning

We describe the applications in the following:

Intersection violation warning

The intersection violation warning (IVW) application warns drivers when they are going to pass over a red light (see Figure 3.7). It is possible to achieve this application by placing a RSU with a traffic light controller, so that the RSU broadcast traffic light information. Vehicles that receive these data can warn the driver about the presence of a red light to avoid accidents in time.

Electronic brake warning

The electronic brake warning (EBW) application reports to the driver that a preceding vehicle has performed a sudden braking. This is useful when the view of the braking vehicle is obstructed by other vehicles. The scenario is shown in Figure 3.8, where vehicle 1, braking violently, produces a message that is sent in broadcast to warn the other vehicles about the dangerous situation.

On-coming traffic warning

The on-coming traffic warning (OTW) application helps the driver during overtaking maneuvers, by providing information about on-coming traffic (see Figure 3.9).
Vehicle stability warning

The vehicle stability warning (VSW) application alerts drivers that they should activate the vehicle stability control system due to the hazardous driving conditions (ice, oil) (see Figure 3.10).

Lane change warning

The lane change warning (LCW) application helps drivers to perform a safe lane change (see Figure 3.11).

3.4 Standards for VANET

In the following, we first describe the basic features of IEEE 802.11 family. Then we describe IEEE 802.11p standard, which is specific for vehicular communications. We conclude with the introduction of IEEE 802.16e standard, that is capable to support mobility at vehicular speeds, and ETSI ITS-G5, that is the European profile of IEEE 802.11p.

3.4.1 Background on IEEE 802.11

IEEE 802.11 standard defines the medium access control (MAC) layer and the physical layer for wireless local area networks (WLANs).

The fundamental element of a WLAN is the basic service set (BSS), that is the set of all stations that can communicate with each other. A station that wants to join an available BSS has to perform the authentication and association processes.

IEEE 802.11 provides two solutions for the MAC layer:

• Point coordination function (PCF): it is only applicable if there is a central coordinating station like an access point.

• Distributed coordination function (DCF): it does not need a centralized control.

It is not possible to have a central coordinating station in Vehicle-to-X communications, so we focus only on DCF.

DCF is based on carrier sense multiple access with collision avoidance (CSMA/CA): the station starts by listening to the channel, and if this is free for a time period called distributed inter-frame space (DIFS), the sender can start to transmit. If the channel is busy or becomes busy during the DIFS, the station has to perform a backoff procedure, that means that the
Figure 3.7: (a) Without IVW, the inattentive driver of vehicle 1 can cause a serious accident due to not stopping at red light. (b) With IVW, the driver of vehicle 1 is warned of red light and stops the vehicle before the intersection.

station must defer its access for a random time (see Figure 3.12). The backoff procedure works as follows:

1. The station extracts a random integer within the contention window (CW);
2. This number is multiplied with the slot time derived from the physical
Figure 3.8: (a) Vehicle 1 has to perform an emergency braking and vehicle 3 does not realize it because it has the view obstructed by the vehicle 2. (b) Without EBW, vehicle 3 cannot react to the delayed braking of vehicle 2 and so there is a collision. (c) With EBW, vehicle 3 is informed of the emergency braking of vehicle 1 and so it has the time to slow down.

Layer in use, and the result is set as the backoff value;

3. The backoff value is decreased only when the channel is free;

4. The station immediately sends when the backoff value is 0.

After the data are transmitted, the sender waits for an acknowledgment (ACK). A backoff procedure must be called if the transmitted packets do not reach the recipient or are incorrect or the ACK does not reach the sender. The size of the CW is doubled for every attempt to send a specific packet up to a maximum value ($CW_{\text{max}}$). The CW is set to the initial value when the packet is correctly received or is discarded because $CW_{\text{max}}$ is reached. Despite the backoff procedure, the hidden node problem can cause a packet collision because two or more stations can transmit simultaneously.
Figure 3.9: (a) Drivers often misjudge velocity and distance of on-coming traffic when they are about to perform an overtake. (b) Without OTW, vehicle 1 causes a collision with both vehicles 2 and 3. (c) With OTW, vehicle 1 is informed about the possible collision and so it decides to perform the overtake only after vehicle 2 has passed.

The physical layer provides several solutions, but we focus on IEEE 802.11a that is used as basic technology to define IEEE 802.11p. IEEE 802.11a uses 20 MHz channels in the 5.2-5.8 GHz frequency band and orthogonal frequency division multiplexing (OFDM) as transmission technology.

### 3.4.2 IEEE 802.11p

The main features of IEEE 802.11p are the following:

- It operates in the 5.9 GHz frequency band;
- It supports vehicular speeds;
Figure 3.10: (a) Vehicle 1 encounters an icy stretch of road and activates its stability control system to maintain the bend. (b) Without VSW, driver 2, that travels at a higher speed, is not able to activate the stability control system and so he loses control and collides with vehicle 3. (c) With VSW, the driver of vehicle 2 is informed of the activation of the stability control system by vehicle 1 and so he slows down to maintain control of the car.

- It operates in a highly dynamic and mobile environment;
- It uses CSMA/CA technique to coordinate medium access.

At the MAC layer, the main difference between IEEE 802.11p and the rest of IEEE 802.11 family is the ability to communicate outside the context of a BSS. This is because authentication and association processes would last too long, not allowing the rapid exchange of safety data.

Another important aspect for vehicular communications is the possibility to assign a higher priority to safety messages. For this purpose, IEEE 802.11p exploits IEEE 802.11e introducing enhanced distributed channel access (EDCA) to provide quality of service (QoS). EDCA is used for medium access (see Figure 3.13) and is an enhanced version of DCF where four access categories (AC) are defined as follows:
Figure 3.11: (a) Vehicle 1 wants to perform a lane change. (b) Without LCW, vehicle 1 does not realize that vehicle 2 is approaching fast and proceeds with the lane change, causing an accident. (c) With LCW, vehicle 1 is informed of the arrival of vehicle 2 and so it waits for vehicle 2 to pass before performing the lane change.

- An access category index (ACI) identifies each AC.
- ACI of 0 identifies regular access, ACI of 1 identifies non-prior traffic and ACIs of 2 and 3 are reserved for prioritized safety messages.
- Each AC has its queue and its parameters.
- The arbitration inter-frame space (AIFS) replaces the fixed DIFS time.
- The higher is priority, the shorter is AIFS.
- CW values are different for each AC.
The physical layer is based on IEEE 802.11a, with some changes to make it suitable for vehicular environment. Like IEEE 802.11a, it uses OFDM as transmission technology for its robustness against fast fading caused by the high relative mobility of vehicles. The possible modulations are BPSK, QPSK, 16-QAM and 64-QAM. Unlike IEEE 802.11a, it uses 10 MHz channels to account for the increased Doppler and delay spreads which could lead to inter-symbol interference (ISI) and inter-carrier interference (ICI), challenging the reception of packets. The data rates are 3, 4.5, 6, 9, 12, 18, 24 and 27 Mbit/s and are halved compared to IEEE 802.11a. These are obtained by varying modulation scheme and code rate. The data rate used for safety applications is 6 Mbit/s, since the higher data rates have higher error rates.

**Other MAC approaches**

CSMA/CA does not guarantee channel access in a finite time and so is not suitable for real-time traffic. Real-time communications are characterized by the two following parameters:

- **Deadline**: a missed deadline can have serious consequences on the system.
- **Reliability**: a message must arrive with a certain error probability.

Safety systems to avoid traffic accidents are classified as real-time systems. MAC methods can be divided in two classes as follows:

- **Deterministic**: in the worst case, channel access delay is finite.
- **Nondeterministic**: it is not suitable for real-time data traffic.
TDMA, FDMA and CDMA are deterministic and suitable for real-time data traffic but they require a central coordinator to manage the resources, which is not present in VANETs. On the other hand, CSMA/CA is implemented in decentralized ad-hoc networks but is also nondeterministic.

In [10] it is proposed to use the self-organizing time division multiple access (STDMA) algorithm, that is part of the automatic identification system (AIS), that is a standard used in the shipping industry. STDMA is a decentralized scheme where the synchronization among the nodes must be done through a global navigation satellite system, such as GPS. In order to establish a mutual awareness, ships have to periodically transmit heartbeat messages which contain information about current position and direction. For this purpose, STDMA divides time into frames of fixed duration, which in turn are divided into equally sized slots. The operations performed by each ship to determine the slots for the transmission of heartbeat messages are the following:

1. Nodes determine a report rate $r$, that is the number of heartbeat mes-
sages in one frame, and so the number of slots required.

2. Each node listens to the channel for the duration of one frame to determine the current usage of slots.

3. Each node selects r unused random slots per frame so that they are equally distributed in time.

4. The slots used by the furthest ship are selected if there are no unused slots.

5. Each reserved slot is used for three to eight frames, and then the same operations must be performed to select new slots.

STDMA is also able to support vehicular communications if fading is not considered. Unfortunately, vehicular environment is subject to fading and it is still unknown whether STDMA can properly work in this situation.

**Congestion control**

Vehicular systems are based on the exchange of two types of messages:

- Beacons: they are periodically broadcasted and contain information on the status of the vehicle.

- Emergency messages: They are transmitted to inform surrounding vehicles when dangerous situations are detected.

Methods to control and limit the load on the radio channel are required, since the transmission of periodic beacons can already saturate the channel. Congestion control algorithms reduce transmission power and beacon generation rate of all node because:

- A high beacon generation rate can increase the accuracy of the information, but can lead to saturate the medium.

- A high transmission power can reach long distances, but increases the level of interference with other transmissions.

Congestion control techniques for vehicular communications can be grouped in two categories:

- Reactive congestion control: measures to reduce the load on the channel are taken only after a congested situation has occurred.
• Proactive congestion control: it uses models to estimate transmission parameters that do not lead to congested situations.

An example of proactive congestion control is the distributed fair-power adjustment for vehicular environments (D-FPAV), that performs congestion control by varying the transmission power of the nodes based on the estimated application layer traffic and on the number of vehicles observed in the surrounding.

3.4.3 IEEE 802.16e

IEEE 802.16e is another standard suitable for vehicular communications and is better known as mobile WiMAX. The main features are the wide coverage and the support of vehicular speeds up to 120 km/h in the 5 GHz band. The access method is based on scalable orthogonal frequency division multiple access (SOFDMA), that dynamically shares the radio bandwidth between all users. Adaptive modulation and coding (AMC) is another technique used in IEEE 802.16e, where modulation and coding scheme are dynamically adapted to the channel conditions in order to obtain a good reliability and the highest spectral efficiency.

An advantage over IEEE 802.11p is that stations have to compete to get access only when they want to join the network, keeping the obtained slot until they leave the network. The slot time can be decreased or increased according to channel load and QoS parameters.

Further studies about the application of this standard to vehicular networks are still necessary.

3.4.4 ETSI ITS-G5

As mentioned in [18], ETSI ITS-G5 is the European standard for vehicular networks and includes features of IEEE 802.11 and IEEE 802.11p:

• Stations can communicate without joining a BSS, avoiding the delays that authentication and association processes would produce.

• Stations must always be able to receive data on the control channel when they are not transmitting.

• The medium access technique is CSMA/CA.

• The physical layer uses OFDM.
As exposed in this chapter, standards for VANET need further studies because all of them present different problems. In the next chapter we introduce linear programming that is necessary to understand the new scheduling algorithm introduced in this thesis.
Chapter 4

Mathematical background

In this chapter we give an overview on linear programming. Then, in the following chapters, we propose the application of linear programming to a scheduling problem for real-time communication control in ITS.

4.1 Linear programming

A linear programming problem can be defined as the problem of maximizing or minimizing a linear function subject to linear constraints.

It is possible to describe two classes of problems called the standard maximum problem and the standard minimum problem.

The standard maximum problem can be defined as follows: find $x_1, \ldots, x_n$ to maximize

$$ c_1 x_1 + \ldots + c_n x_n $$

subject to

$$ a_{11} x_1 + a_{12} x_2 + \ldots + a_{1n} x_n \leq b_1 $$
$$ a_{21} x_1 + a_{22} x_2 + \ldots + a_{2n} x_n \leq b_2 $$
$$ \vdots $$
$$ a_{m1} x_1 + a_{m2} x_2 + \ldots + a_{mn} x_n \leq b_m $$

and

$$ x_1 \geq 0, x_2 \geq 0, \ldots, x_n \geq 0. $$

The standard minimum problem can be defined as follows: find $y_1, \ldots, y_m$ to minimize

$$ y_1 b_1 + \ldots + y_m b_m $$
subject to

\[ y_1 a_{11} + y_2 a_{21} + \ldots + y_m a_{m1} \geq c_1 \]
\[ y_1 a_{12} + y_2 a_{22} + \ldots + y_m a_{m2} \geq c_2 \]
\[ \vdots \]
\[ y_1 a_{1n} + y_2 a_{2n} + \ldots + y_m a_{mn} \geq c_n \]

and

\[ y_1 \geq 0, y_2 \geq 0, \ldots, y_m \geq 0. \]

These two problems can be reformulated giving an \( m \)-vector, \( b = (b_1, \ldots, b_m)^T \), an \( n \)-vector \( c = (c_1, \ldots, c_n)^T \), and an \( m \times n \) matrix of real numbers,

\[
A = \begin{pmatrix}
a_{11} & a_{12} & \cdots & a_{1n} \\
a_{21} & a_{22} & \cdots & a_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
a_{m1} & a_{m2} & \cdots & a_{mn}
\end{pmatrix}
\]

The standard maximum problem becomes: find an \( n \)-vector, \( x = (x_1, \ldots, x_n)^T \), to maximize

\[ c^T x \]

subject to

\[ Ax \leq b \]

and

\[ x \geq 0. \]

The standard minimum problem becomes: find an \( m \)-vector, \( y = (y_1, \ldots, y_m)^T \), to minimize

\[ y^T b \]

subject to

\[ y^T A \geq c^T \]

and

\[ y \geq 0. \]

The function that has to be maximized or minimized is called objective function, the constraints \( x \geq 0 \) or \( y \geq 0 \) are called nonnegativity constraints and the other constraints are called main constraints.

A vector \( x \) or \( y \) is said feasible if it satisfies the constraints and the set of the feasible vectors is called constraint set.

A linear programming problem can be:
• Feasible: the constraint set is not empty;
• Infeasible: the constraint set is empty.

A feasible problem can be:
• Unbounded: the objective function can assume arbitrarily large values;
• Bounded: the objective function can not assume arbitrarily large values.

The value of a bounded feasible maximum (minimum) problem is the maximum (minimum) value of the objective function and the feasible vector at which the objective function achieves this value is called optimal.

All linear programming problems can be converted to a standard form using the following techniques:
• It is possible to turn a minimum problem into a maximum problem by multiplying the objective function by -1;
• Main constraints of the form \( \sum_{j=1}^{n} a_{ij} x_j \geq b_i \) can be changed into \( \sum_{j=1}^{n} (-a_{ij}) x_j \leq -b_i \);
• It is possible to remove an equality constraint by solving the constraint for some \( x_j \) for which \( a_{ij} \neq 0 \) and substituting the solution into the objective function and into the other constraints wherever \( x_j \) appears;
• It is possible to replace a variable that is not restricted to be nonnegative with the difference of two nonnegative variables.

4.1.1 Duality

The dual of the standard maximum problem, maximize \( c^T x \) subject to \( Ax \leq b \) and \( x \geq 0 \), is the standard minimum problem, minimize \( y^T b \) subject to \( y^T A \geq c^T \) and \( y \geq 0 \). In the same way, the dual of the standard minimum problem is the standard maximum problem, so these two problems are said to be duals.

The following theorems and corollaries point out the relation between a standard problem and its dual.

**Theorem 1** [11] If \( x \) is feasible for the standard maximum problem and if \( y \) is feasible for its dual, then
\[
c^T x \leq y^T b.
\]
Corollary 1 [11] If a standard problem and its dual are both feasible, then both are bounded feasible.

Corollary 2 [11] If there exist feasible \( \mathbf{x}^* \) and \( \mathbf{y}^* \) for a standard maximum problem and its dual such that \( \mathbf{c}^T \mathbf{x}^* = \mathbf{y}^T \mathbf{b} \), then both are optimal for their respective problems.

Theorem 2 (The Duality Theorem) [11] If a standard linear programming problem is bounded feasible, then so is its dual, their values are equal, and there exist optimal vectors for both problems.

Theorem 3 (The Equilibrium Theorem) [11] Let \( \mathbf{x}^* \) and \( \mathbf{y}^* \) be feasible vectors for a standard maximum problem and its dual respectively. Then \( \mathbf{x}^* \) and \( \mathbf{y}^* \) are optimal if, and only if,

\[
y^*_i = 0 \text{ } \forall \text{ } i \text{ for which } \sum_{j=1}^{n} a_{ij}x^*_j < b_i
\]

and

\[
x^*_j = 0 \text{ } \forall \text{ } j \text{ for which } \sum_{i=1}^{m} y^*_i a_{ij} > c_j.
\]

4.1.2 Graphical method

We can solve a linear programming problem using the graphical method as shown in the example proposed in [11]: find \( x_1 \) and \( x_2 \) that maximize \( x_1 + x_2 \) subject to \( x_1 \geq 0, \) \( x_2 \geq 0, \) and

\[
\begin{align*}
x_1 + 2x_2 & \leq 4 \\
4x_1 + 2x_2 & \leq 12 \\
-x_1 + x_2 & \leq 1
\end{align*}
\]

We start graphing the constraint set, that is the set of points in the plane that satisfies the constraints, and then finding the point of this set that maximizes the value of the objective function (see Figure 4.1).

We are looking for the point \((x_1, x_2)\) of the constraint set that maximizes \( x_1 + x_2 \). If we consider the line \( x_1 + x_2 = k \), we know that \( k \), and so \( x_1 + x_2 \), increases as we move far from the origin up and to the right. Therefore, we are looking for the line \( x_1 + x_2 = k \) that is farther from the origin and still touches the constraint set. The optimal point is \((x_1, x_2) = (8/3, 2/3)\) and corresponds to the intersection of the lines \( x_1 + 2x_2 = 4 \) and \( 4x_1 + 2x_2 = 12 \). The value of the objective function is 10/3.
If the constraint set is bounded, the objective function always achieves its maximum (minimum) value at a corner point of the constraint set.

The graphical method is useful if there are only two variables.

4.1.3 The simplex method

The simplex method is used to solve linear programming problems. This method can be formalized in the simplex tableau, that can be written as follows:

\[
\begin{array}{cccc|c}
  x_1 & x_2 & \cdots & x_n & \\
  y_1 & a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\
  y_2 & a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\
  \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
  y_m & a_{m1} & a_{m2} & \cdots & a_{mn} & b_m \\
  -c_1 & -c_2 & \cdots & -c_n & 0
\end{array}
\]

We can use the pivoting rules to interchange one variable of the first row with one variable of the first column on the left. To indicate that we are going to interchange two variables, we circle the corresponding entry \(a_{ij}(\neq 0)\) that is called the pivot. The pivoting rules can be summarized as follows:

\[
\begin{array}{cc|cc}
Q & r & 1/p & r/p \\
-c & q & -c/p & q - (rc/p)
\end{array}
\]

We have to pivot until all entries in the last row and column are nonnegative except the corner. The value of the linear programming problem and
its dual is in the lower right corner. The solution of the minimum problem is obtained by assuming that the $y_i$ that are in the first column on the left are equal to zero, and the $y_i$ that are in the first row are equal to the corresponding entries in the last row. The solution of the maximum problem is obtained by assuming that the $x_j$ that are in the first row are equal to zero, and the $x_j$ that are in the first column on the left are equal to the corresponding entries in the last column.

The simplex method tells us the points to choose as pivots to approach the solution.

The pivoting rules for the simplex method from the point of view of the maximum problem are the following [11]:

1. $b \geq 0$: Take any column with last entry negative, say column $j_0$ with $-c_{j_0} < 0$. Among those $i$ for which $a_{i,j_0} > 0$, choose that $i_0$ for which the ratio $b_i/a_{i,j_0}$ is smallest. If there are ties, choose any such $i_0$. Pivot around $a_{i_0,j_0}$.

2. Some $b_i$ are negative: Take the first negative $b_i$, say $b_k < 0$ (where $b_1 \geq 0, ... , b_{k-1} \geq 0$). Find any negative entry in row $k$, say $a_{k,j_0} < 0$. The pivot will be in column $j_0$. Compare $b_k/a_{k,j_0}$ and the $b_i/a_{i,j_0}$ for which $b_i \geq 0$ and $a_{i,j_0} > 0$, and choose $i_0$ for which this ratio is smallest ($i_0$ may be equal to $k$). You may choose any such $i_0$ if there are ties. Pivot on $a_{i_0,j_0}$.

The pivoting rules for the simplex method from the point of view of the minimum problem are the following [11]:

1. $-c \geq 0$: Take any row with last entry negative, say $b_{i_0} < 0$. Among those $j$ for which $a_{i_0,j} < 0$, choose that $j_0$ for which the ratio $-c_j/a_{i_0,j}$ is closest to zero. Pivot on $a_{i_0,j_0}$.

2. Some $-c_j$ are negative: Take the first negative $-c_j$, say $-c_k < 0$ (where $-c_1 \geq 0, ... , -c_{k-1} \geq 0$). Find any positive entry in column $k$, say $a_{i_0,k} > 0$. Compare $-c_k/a_{i_0,k}$ and those $-c_j/a_{i_0,j}$ for which $-c_j \geq 0$ and $a_{i_0,j} < 0$, and choose $j_0$ for which this ratio is closest to zero ($j_0$ may be $k$). Pivot on $a_{i_0,j_0}$.

4.1.4 General problem

Now we consider the general form of the linear programming problem, where is possible to find equality constraints and unrestricted variables.
The general maximum problem can be defined as follows: find $x_1, \ldots, x_n$ to maximize
$$c^T x$$
subject to
$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i \quad \text{for } i = 1, \ldots, k$$
$$\sum_{j=1}^{n} a_{ij} x_j = b_i \quad \text{for } i = k + 1, \ldots, m$$
and
$$x_j \geq 0 \quad \text{for } j = 1, \ldots, l$$
$$x_j \text{ unrestricted} \quad \text{for } j = l + 1, \ldots, n.$$

The dual of this problem is the general minimum problem that can be defined as follows: find $y_1, \ldots, y_m$ to minimize
$$y^T b$$
subject to
$$\sum_{i=1}^{m} y_i a_{ij} \geq c_j \quad \text{for } j = 1, \ldots, l$$
$$\sum_{i=1}^{m} y_i a_{ij} = c_j \quad \text{for } j = l + 1, \ldots, n$$
and
$$y_i \geq 0 \quad \text{for } i = 1, \ldots, k$$
$$y_i \text{ unrestricted} \quad \text{for } i = k + 1, \ldots, m.$$

It is possible to notice that a strict equality in one program corresponds to an unrestricted variable in the dual.

We can use the simplex method to find the solution of the general problem, but we have to proceed as follows:

1. All equality constraints are pivoted and deleted;
2. All unrestricted variables are pivoted and deleted;
3. We can use the rules of the simplex method to find the optimal solution.

In the next chapter we introduce the new scheduling algorithm for real-time traffic.
Chapter 5

Scheduling algorithm

Scheduling algorithms are necessary to optimally allocate resources to multiple users. In this chapter we focus on a new scheduling algorithm for ITS.

In Section 5.1 we explain the scheduling algorithm for real-time services proposed in [13] that we use as starting point for our analysis. In Section 5.2 we introduce our new scheduling algorithm.

5.1 Scheduling algorithm for real-time services

In [13] it is proposed a scheduling algorithm for real-time services based on average delay requirements. It is considered a scenario where $K$ users are connected to an access point over wireless links. It is assumed time division multiple access (TDMA) for the uplink. An optimization approach is used to maximize the following objective function:

$$\sum_{k=1}^{K} U_{RT,k}(d_k)$$

where $d_k$ is the average queueing delay and the function $U_{RT,k}($·$)$ must be chosen concave and monotonically decreasing because the utility for user $k$ decreases as $d_k$ increases. Denoting with $q_k[n]$ the queue size in bits of user $k$ at the beginning of slot $n$, the queue size of user $k$ at time slot $n + 1$ is the following:

$$q_k[n + 1] = q_k[n] - \min\{\tau_k(h[n])r_k(h_k[n])T_s, q_k[n]\} + \alpha_k[n]$$

where $\tau_k(h[n])r_k(h_k[n])$ is the departure rate, $h[n] = [h_1[n], ..., h_K[n]]^T$ is the vector of channel gains between the access point and the $K$ users, $T_s$ is the
time slot duration and $\alpha_k[n]$ is the number of arriving bits at slot $n$. The average queueing delay of user $k$ at time slot $n+1$ is the following:

$$d_{k}[n+1] = d_k[n] + \beta_n(\alpha_k^{-1}(q_k[n] + \alpha_k[n]) - \min\{\tau_k(h[n])r_k(h_k[n])T_s, q_k[n]\}) - d_k[n]$$

where $\beta_n \in (0, 1)$ implements a forgetting factor in the averaging. Denoting with $\lambda[n] = [\lambda_1[n], ..., \lambda_K[n]]^T$ the Lagrange multiplier vector corresponding to the average delay constraints at time slot $n$ and after a first-order approximation of Taylor’s expansion, the following linear programming problem is obtained:

$$\max_{\tau(h[n])} \sum_{k=1}^{K} (-U_{RT,k}'(d_k[n]) + \lambda_k[n])\tau_k(h[n])r_k(h_k[n])$$

s.to

$$\sum_{k=1}^{K} \tau_k(h[n]) \leq 1$$

$$\tau_k(h[n])r_k(h_k[n]) \leq \frac{q_k[n]}{T_s}, \forall k.$$

We use this scheduling algorithm as starting point for our analysis.

### 5.2 Proposed scheduling algorithm

We propose to use a TDMA MAC method where the roadside unit has the tasks to estimate the channel conditions and assign fractions of time slot to users.

We have considered a multi-hop scenario constituted by eight configurations both for uplink and downlink. By uplink, we mean the link from a vehicle up to a roadside unit, whereas by downlink, we mean the link from a roadside unit down to a vehicle. We have separately considered uplink and downlink, because in downlink we have to take the queues of the roadside units into account. Each configuration is constituted by two vehicles, a roadside unit and different links between them that correspond to the fractions of time slot to assign to users. We have formulated and solved a linear programming problem for each configuration. The optimal solution is both the duration of the fractions of time slot and the configuration. To take the optimal configuration we have to solve the linear programming problems for all the configurations, and we have to pick the configuration with the best cost function value. As all the linear programming problems formulated are minimization problems, the optimal configuration is the one with the minimum cost function value. The duration of the fraction of time slot depends on the
channel gain, so a higher channel gain corresponds to a higher transmission time whereas a lower channel gain corresponds to a lower transmission time.

We can make the following assumptions:

- \( J \) is the number of users;
- \( U_{RT,j}(\cdot) \) must be chosen concave and monotonically decreasing;
- \( T_s \) is the time slot duration;
- \( n \) is the time slot index;
- \( \tau_k(h[n]) \) is the fraction of time slot that depends on the channel gain;
- \( K \) is the number of fractions of time slot;
- \( h_i[n] \) is the channel gain;
- \( h[n] = [h_1[n], ..., h_I[n]]^T \) is the vector of channel gains;
- \( q_j[n] \) is the queue size in bits at the beginning of slot \( n \) (vehicle);
- \( q^*_j[n] \) is the queue size in bits at the beginning of slot \( n \) (roadside unit);
- \( d_j[n] \) is the average queueing delay (vehicle);
- \( d^*_j[n] \) is the average queueing delay (roadside unit);
- \( r_i(h_i[n]) \) is the transmission rate that depends on the channel gain;
- \( r_i(h_i[n]) = \log_2(1 + h_i[n]P) \) is the Shannon's formula for the transmission rate;
- \( P \) is the fixed transmit power;
- \( \tau_k(h[n])r_i(h_i[n]) \) is the departure rate;
- \( q_{\text{max}} \) is the maximum queue capacity in bits.

5.2.1 Uplink

The uplink scenario is shown in Figure 5.1. We consider only two vehicles even if it is possible to extend this study to more users. As we can see, both vehicles have their own queue (vehicle 1 with \( q_1[n] \) and vehicle 2 with \( q_2[n] \)) and there are three different channel gains (\( h_1[n], h_2[n], h_3[n] \)).
Configuration 1

In this configuration the two vehicles transmit directly to the roadside unit, as shown in Figure 5.2.
The queue size of vehicle 1 at time slot $n+1$ is the following:

$$q_1[n+1] = q_1[n] - \tau_1(h[n])r_1(h_1[n])T_s$$

The queue size of vehicle 2 at time slot $n+1$ is the following:

$$q_2[n+1] = q_2[n] - \tau_2(h[n])r_2(h_2[n])T_s$$

The linear programming problem for configuration 1 is the following:

$$\min_{\tau(h[n])} \quad U'_{RT,1}(d_1[n])\tau_1(h[n])r_1(h_1[n]) + U'_{RT,2}(d_2[n])\tau_2(h[n])r_2(h_2[n])$$

s.t.

1. \( \tau_1(h[n]) \geq 0 \) \hspace{1cm} (5.1a)
2. \( \tau_2(h[n]) \geq 0 \) \hspace{1cm} (5.1b)
3. \( \tau_1(h[n]) + \tau_2(h[n]) \leq T_s \) \hspace{1cm} (5.1c)
4. \( \tau_1(h[n])r_1(h_1[n]) \leq \frac{q_1[n]}{T_s} \) \hspace{1cm} (5.1d)
5. \( \tau_2(h[n])r_2(h_2[n]) \leq \frac{q_2[n]}{T_s} \) \hspace{1cm} (5.1e)

The meanings of the constraints are the following:

- (5.1a) and (5.1b) mean that the fractions of time slot must be positive or zero;
- (5.1c) means that the sum of the fractions of time slot must be lower or equal to time slot duration;
- (5.1d) and (5.1e) mean that vehicles cannot transmit more data than those in their queue;
- In this case we do not need constraints for congestion control.

On certain situations of the channel gains, fast numerical techniques can be used, such as Fast-Lipschitz Optimization [24], but in the following we want to find the analytical solution using the simplex method as discussed in Chapter 4. After having multiplied the objective function by -1 to obtain a standard maximum problem, we can write the simplex tableau as follows:
After using the pivoting rules for the simplex method, we obtain the following solutions:

1. If $T_s < \frac{q_1[n]}{T_s r_1(h_1[n])}$:

   (a) If $U_{RT,2}(d_2[n]) r_2(h_2[n]) \geq U_{RT,1}(d_1[n]) r_1(h_1[n])$:

      \[
      \begin{align*}
      \tau_1(h[n]) &= T_s \\
      \tau_2(h[n]) &= 0 \\
      v &= -T_s U'_{RT,1}(d_1[n]) r_1(h_1[n])
      \end{align*}
      \]

   (b) If $U_{RT,2}(d_2[n]) r_2(h_2[n]) < U_{RT,1}(d_1[n]) r_1(h_1[n])$:

      i. If $T_s < \frac{q_2[n]}{T_s r_2(h_2[n])}$:

      \[
      \begin{align*}
      \tau_1(h[n]) &= 0 \\
      \tau_2(h[n]) &= T_s \\
      v &= -T_s U'_{RT,2}(d_2[n]) r_2(h_2[n])
      \end{align*}
      \]

      ii. If $T_s > \frac{q_2[n]}{T_s r_2(h_2[n])}$:

      \[
      \begin{align*}
      \tau_1(h[n]) &= T_s - \frac{q_2[n]}{T_s r_2(h_2[n])} \\
      \tau_2(h[n]) &= \frac{q_2[n]}{T_s r_2(h_2[n])} \\
      v &= -T_s U'_{RT,1}(d_1[n]) r_1(h_1[n]) + \frac{q_2[n] U'_{RT,1}(d_1[n]) r_1(h_1[n])}{T_s r_2(h_2[n])} \\
      &\quad - \frac{q_2[n] U'_{RT,2}(d_2[n])}{T_s}
      \end{align*}
      \]

2. If $T_s > \frac{q_1[n]}{T_s r_1(h_1[n])}$:
(a) If $T_s < \frac{q_1[n]}{T_{sr_1(h_1[n])}} + \frac{q_2[n]}{T_{sr_2(h_2[n])}}$:

$$
\tau_1(h[n]) = \frac{q_1[n]}{T_{sr_1(h_1[n])}} \\
\tau_2(h[n]) = T_s - \frac{q_1[n]}{T_{sr_1(h_1[n])}} \\
v = -T_sU_{RT,2}'(d_2[n])r_2(h_2[n]) + \frac{q_1[n]U_{RT,2}'(d_2[n])r_2(h_2[n])}{T_s r_1(h_1[n])} - \frac{q_1[n]U_{RT,1}'(d_1[n])}{T_s} \tag{10}
$$

(b) If $T_s > \frac{q_1[n]}{T_{sr_1(h_1[n])}} + \frac{q_2[n]}{T_{sr_2(h_2[n])}}$:

$$
\tau_1(h[n]) = \frac{q_1[n]}{T_{sr_1(h_1[n])}} \\
\tau_2(h[n]) = \frac{q_2[n]}{T_{sr_2(h_2[n])}} \\
v = -T_sU_{RT,2}'(d_2[n])r_2(h_2[n]) - \frac{q_1[n]U_{RT,1}'(d_1[n]) - q_2[n]U_{RT,2}'(d_2[n])}{T_s} \tag{11}
$$

**Configuration 2**

In this configuration vehicle 1 transmits its data to vehicle 2 and then vehicle 2 transmits to the roadside unit, as shown in Figure 5.3.

The queue size of vehicle 1 at time slot $n + 1$ is the following:

$$
q_1[n + 1] = q_1[n] - \tau_1(h[n])r_3(h_3[n])T_s
$$

The queue size of vehicle 2 at time slot $n + 1$ is the following:

$$
q_2[n + 1] = q_2[n] + \tau_1(h[n])r_3(h_3[n])T_s - \tau_2(h[n])r_2(h_2[n])T_s
$$
The linear programming problem for configuration 2 is the following:

\[
\begin{align*}
\min_{\tau(h[n])} & \quad U_{RT,1}'(d_1[n])\tau_1(h[n])r_3(h_3[n]) + U_{RT,2}'(d_2[n])\tau_2(h[n])r_2(h_2[n]) \\
\text{s.t} & \quad \tau_1(h[n]) \geq 0 \quad (5.3a) \\
& \quad \tau_2(h[n]) \geq 0 \quad (5.3b) \\
& \quad \tau_1(h[n]) + \tau_2(h[n]) \leq T_s \quad (5.3c) \\
& \quad \tau_1(h[n])r_3(h_3[n]) \leq \frac{q_1[n]}{T_s} \quad (5.3d) \\
& \quad -\tau_1(h[n])r_3(h_3[n]) + \tau_2(h[n])r_2(h_2[n]) \leq \frac{q_2[n]}{T_s} \quad (5.3e) \\
& \quad \tau_1(h[n])r_3(h_3[n]) \leq \frac{q_{\text{max}} - q_2[n]}{T_s} \quad (5.3f)
\end{align*}
\]

The meanings of the constraints are the following:

- (5.3a) and (5.3b) mean that the fractions of time slot must be positive or zero;
- (5.3c) means that the sum of the fractions of time slot must be lower or equal to time slot duration;
- (5.3d) and (5.3e) mean that vehicles cannot transmit more data than those in their queue;
• (5.3f) is a constraint for congestion control that comes from:

\[ q_2[n] + \tau_1(h[n])r_3(h_3[n])T_s \leq q_{\text{max}}, \]

that means that it is not possible to exceed the maximum queue capacity.

**Configuration 3**

In this configuration vehicle 2 transmits to the roadside unit, then vehicle 1 transmits its data to vehicle 2 and finally vehicle 2 transmits again to the roadside unit, as shown in Figure 5.4.

![Figure 5.4: Uplink - Configuration 3](image)

The queue size of vehicle 1 at time slot \( n + 1 \) is the following:

\[ q_1[n + 1] = q_1[n] - \tau_2(h[n])r_3(h_3[n])T_s \]

The queue size of vehicle 2 at time slot \( n + 1 \) is the following:

\[ q_2[n + 1] = q_2[n] - \tau_1(h[n])r_2(h_2[n])T_s + \tau_2(h[n])r_3(h_3[n])T_s - \tau_3(h[n])r_2(h_2[n])T_s \]
The linear programming problem for configuration 3 is the following:

\[
\min_{\tau(h[n])} \quad U'_{RT,2}(d_2[n])\tau_1(h[n])r_2(h_2[n]) + U'_{RT,1}(d_1[n])\tau_2(h[n])r_3(h_3[n]) \\
+ U'_{RT,2}(d_2[n])\tau_3(h[n])r_2(h_2[n])
\]

s.t
\[
\tau_1(h[n]) \geq 0 \quad (5.4a) \\
\tau_2(h[n]) \geq 0 \quad (5.4b) \\
\tau_3(h[n]) \geq 0 \quad (5.4c) \\
\tau_1(h[n]) + \tau_2(h[n]) + \tau_3(h[n]) \leq T_s \quad (5.4d) \\
\tau_1(h[n])r_2(h_2[n]) \leq \frac{q_2[n]}{T_s} \quad (5.4e) \\
\tau_2(h[n])r_3(h_3[n]) \leq \frac{q_1[n]}{T_s} \quad (5.4f) \\
\tau_1(h[n])r_2(h_2[n]) - \tau_2(h[n])r_3(h_3[n]) + \tau_3(h[n])r_2(h_2[n]) \leq \frac{q_2[n]}{T_s} \quad (5.4g) \\
-\tau_1(h[n])r_2(h_2[n]) + \tau_2(h[n])r_3(h_3[n]) \leq \frac{q_{\text{max}} - q_2[n]}{T_s} \quad (5.4h)
\]

The meanings of the constraints are the following:

- (5.4a), (5.4b) and (5.4c) mean that the fractions of time slot must be positive or zero;
- (5.4d) means that the sum of the fractions of time slot must be lower or equal to time slot duration;
- (5.4e), (5.4f) and (5.4g) mean that vehicles cannot transmit more data than those in their queue;
- (5.4h) is a constraint for congestion control that comes from:

\[
q_2[n] - \tau_1(h[n])r_2(h_2[n])T_s + \tau_2(h[n])r_3(h_3[n])T_s \leq q_{\text{max}},
\]

that means that it is not possible to exceed the maximum queue capacity.
Configuration 4

In this configuration vehicle 1 transmits to the roadside unit, then vehicle 1 transmits to vehicle 2 and finally vehicle 2 transmits to the roadside unit, as shown in Figure 5.5.

The queue size of vehicle 1 at time slot $n + 1$ is the following:

$$q_1[n + 1] = q_1[n] - \tau_1(h[n])r_1(h_1[n])T_s - \tau_2(h[n])r_3(h_3[n])T_s$$

The queue size of vehicle 2 at time slot $n + 1$ is the following:

$$q_2[n + 1] = q_2[n] + \tau_2(h[n])r_3(h_3[n])T_s - \tau_3(h[n])r_2(h_2[n])T_s$$
The linear programming problem for configuration 4 is the following:

\[
\min_{\tau(h[n])} U_{RT,1}(d_1[n])\tau_1(h[n])r_1(h_1[n]) + U_{RT,1}'(d_1[n])\tau_2(h[n])r_3(h_3[n]) \\
+ U_{RT,2}(d_2[n])\tau_3(h[n])r_2(h_2[n])
\]

s.t.
\[
\begin{align*}
&\tau_1(h[n]) \geq 0 \\
&\tau_2(h[n]) \geq 0 \\
&\tau_3(h[n]) \geq 0 \\
&\tau_1(h[n]) + \tau_2(h[n]) + \tau_3(h[n]) \leq T_s \\
&\tau_1(h[n])r_1(h_1[n]) \leq \frac{q_1[n]}{T_s} \\
&\tau_1(h[n])r_1(h_1[n]) + \tau_2(h[n])r_3(h_3[n]) \leq \frac{q_1[n]}{T_s} \\
&-\tau_2(h[n])r_3(h_3[n]) + \tau_3(h[n])r_2(h_2[n]) \leq \frac{q_2[n]}{T_s} \\
&\tau_2(h[n])r_3(h_3[n]) \leq \frac{q_{\text{max}} - q_2[n]}{T_s}.
\end{align*}
\]

The meanings of the constraints are the following:

- (5.5a), (5.5b) and (5.5c) mean that the fractions of time slot must be positive or zero;
- (5.5d) means that the sum of the fractions of time slot must be lower or equal to time slot duration;
- (5.5e), (5.5f) and (5.5g) mean that vehicles cannot transmit more data than those in their queue;
- (5.5h) is a constraint for congestion control that comes from:
\[
q_2[n] + \tau_2(h[n])r_3(h_3[n])T_s \leq q_{\text{max}},
\]

that means that it is not possible to exceed the maximum queue capacity.
**Configuration 5**

In this case we have four fractions of time slot and the scenario is similar to configuration 4 except that vehicle 2 transmits to the roadside unit for the last two consecutive fractions of time slot, as shown in Figure 5.6.

![Configuration 5 Diagram](image)

**Figure 5.6: Uplink - Configuration 5**

The queue size of vehicle 1 at time slot $n + 1$ is the following:

$$q_1[n + 1] = q_1[n] - \tau_1(h[n])r_1(h_1[n])T_s - \tau_2(h[n])r_3(h_3[n])T_s$$

The queue size of vehicle 2 at time slot $n + 1$ is the following:

$$q_2[n+1] = q_2[n] + \tau_2(h[n])r_3(h_3[n])T_s - \tau_3(h[n])r_2(h_2[n])T_s - \tau_4(h[n])r_2(h_2[n])T_s$$
The linear programming problem for configuration 5 is the following:

\[
\begin{align*}
\min_{\tau(h[n])} & \quad U'_{RT,1}(d_1[n])\tau_1(h[n])r_1(h_1[n]) + U'_{RT,1}(d_1[n])\tau_2(h[n])r_3(h_3[n]) \\
& \quad + U'_{RT,2}(d_2[n])\tau_3(h[n])r_2(h_2[n]) + U'_{RT,2}(d_2[n])\tau_4(h[n])r_2(h_2[n]) \\
\text{s.t} & \quad \tau_1(h[n]) \geq 0 \quad (5.6a) \\
& \quad \tau_2(h[n]) \geq 0 \quad (5.6b) \\
& \quad \tau_3(h[n]) \geq 0 \quad (5.6c) \\
& \quad \tau_4(h[n]) \geq 0 \quad (5.6d) \\
& \quad \tau_1(h[n]) + \tau_2(h[n]) + \tau_3(h[n]) + \tau_4(h[n]) \leq T_s \quad (5.6e) \\
& \quad \tau_1(h[n])r_1(h_1[n]) \leq \frac{q_1[n]}{T_s} \quad (5.6f) \\
& \quad \tau_1(h[n])r_1(h_1[n]) + \tau_2(h[n])r_3(h_3[n]) \leq \frac{q_1[n]}{T_s} \quad (5.6g) \\
& \quad -\tau_2(h[n])r_3(h_3[n]) + \tau_3(h[n])r_2(h_2[n]) \leq \frac{q_2[n]}{T_s} \quad (5.6h) \\
& \quad -\tau_2(h[n])r_3(h_3[n]) + \tau_3(h[n])r_2(h_2[n]) + \tau_4(h[n])r_2(h_2[n]) \\
& \quad \leq \frac{q_2[n]}{T_s} \quad (5.6i) \\
& \quad \tau_2(h[n])r_3(h_3[n]) \leq \frac{q_{\text{max}} - q_2[n]}{T_s} \quad (5.6j)
\end{align*}
\]

The meanings of the constraints are the following:

- (5.6a), (5.6b), (5.6c) and (5.6d) mean that the fractions of time slot must be positive or zero;
- (5.6e) means that the sum of the fractions of time slot must be lower or equal to time slot duration;
- (5.6f), (5.6g), (5.6h) and (5.6i) mean that vehicles cannot transmit more data than those in their queue;
- (5.6j) is a constraint for congestion control that comes from:

\[q_2[n] + \tau_2(h[n])r_3(h_3[n])T_s \leq q_{\text{max}},\]

that means that it is not possible to exceed the maximum queue capacity.
**Configuration 6**

In this configuration vehicle 2 transmits its data to vehicle 1 and then vehicle 1 transmits to the roadside unit, as shown in Figure 5.7.

The queue size of vehicle 1 at time slot $n + 1$ is the following:

$$q_1[n + 1] = q_1[n] + \tau_1(h[n])r_3(h_3[n])T_s - \tau_2(h[n])r_1(h_1[n])T_s$$

The queue size of vehicle 2 at time slot $n + 1$ is the following:

$$q_2[n + 1] = q_2[n] - \tau_1(h[n])r_3(h_3[n])T_s$$

The linear programming problem for configuration 6 is the following:

$$\min_{\tau(h[n])} U'_{RT,2}(d_2[n])\tau_1(h[n])r_3(h_3[n]) + U'_{RT,1}(d_1[n])\tau_2(h[n])r_1(h_1[n])$$

s.t.

$$\tau_1(h[n]) \geq 0 \quad (5.7a)$$

$$\tau_2(h[n]) \geq 0 \quad (5.7b)$$

$$\tau_1(h[n]) + \tau_2(h[n]) \leq T_s \quad (5.7c)$$

$$\tau_1(h[n])r_3(h_3[n]) \leq \frac{q_2[n]}{T_s} \quad (5.7d)$$

$$-\tau_1(h[n])r_3(h_3[n]) + \tau_2(h[n])r_1(h_1[n]) \leq \frac{q_1[n]}{T_s} \quad (5.7e)$$

$$\tau_1(h[n])r_3(h_3[n]) \leq \frac{q_{\text{max}} - q_1[n]}{T_s}. \quad (5.7f)$$
The meanings of the constraints are the following:

- (5.7a) and (5.7b) mean that the fractions of time slot must be positive or zero;
- (5.7c) means that the sum of the fractions of time slot must be lower or equal to time slot duration;
- (5.7d) and (5.7e) mean that vehicles cannot transmit more data than those in their queue;
- (5.7f) is a constraint for congestion control that comes from:

\[ q_1[n] + \tau_1(h[n])r_3(h_3[n])T_s \leq q_{\text{max}}, \]

that means that it is not possible to exceed the maximum queue capacity.

**Configuration 7**

In this configuration vehicle 1 transmits to the roadside unit, then vehicle 2 transmits its data to vehicle 1 and finally vehicle 1 transmits again to the roadside unit, as shown in Figure 5.8.

![Figure 5.8: Uplink - Configuration 7](image)

The queue size of vehicle 1 at time slot \( n+1 \) is the following:

\[ q_1[n+1] = q_1[n] - \tau_1(h[n])r_1(h_1[n])T_s + \tau_2(h[n])r_3(h_3[n])T_s - \tau_3(h[n])r_1(h_1[n])T_s \]
The queue size of vehicle 2 at time slot $n + 1$ is the following:

$$q_2[n + 1] = q_2[n] - \tau_2(h[n])r_3(h_3[n])T_s$$

The linear programming problem for configuration 7 is the following:

$$\min_{\tau(h[n])} U'_{RT,1}(d_1[n])\tau_1(h[n])r_1(h_1[n]) + U'_{RT,2}(d_2[n])\tau_2(h[n])r_3(h_3[n])$$

$$+ U'_{RT,1}(d_1[n])\tau_3(h[n])r_1(h_1[n])$$

s.to

$$\tau_1(h[n]) \geq 0$$ (5.8a)

$$\tau_2(h[n]) \geq 0$$ (5.8b)

$$\tau_3(h[n]) \geq 0$$ (5.8c)

$$\tau_1(h[n]) + \tau_2(h[n]) + \tau_3(h[n]) \leq T_s$$ (5.8d)

$$\tau_1(h[n])r_1(h_1[n]) \leq \frac{q_1[n]}{T_s}$$ (5.8e)

$$\tau_2(h[n])r_3(h_3[n]) \leq \frac{q_2[n]}{T_s}$$ (5.8f)

$$\tau_1(h[n])r_1(h_1[n]) - \tau_2(h[n])r_3(h_3[n]) + \tau_3(h[n])r_1(h_1[n]) \leq \frac{q_1[n]}{T_s}$$ (5.8g)

$$-\tau_1(h[n])r_1(h_1[n]) + \tau_2(h[n])r_3(h_3[n]) \leq \frac{q_{\text{max}} - q_1[n]}{T_s}.$$ (5.8h)

The meanings of the constraints are the following:

- (5.8a), (5.8b) and (5.8c) mean that the fractions of time slot must be positive or zero;

- (5.8d) means that the sum of the fractions of time slot must be lower or equal to time slot duration;

- (5.8e), (5.8f) and (5.8g) mean that vehicles cannot transmit more data than those in their queue;

- (5.8h) is a constraint for congestion control that comes from:

$$q_1[n] - \tau_1(h[n])r_1(h_1[n])T_s + \tau_2(h[n])r_3(h_3[n])T_s \leq q_{\text{max}},$$

that means that it is not possible to exceed the maximum queue capacity.
Configuration 8

In this configuration vehicle 2 transmits to the roadside unit, then vehicle 2 transmits to vehicle 1 and finally vehicle 1 transmits to the roadside unit, as shown in Figure 5.9.

The queue size of vehicle 1 at time slot $n+1$ is the following:

$$q_1[n+1] = q_1[n] + \tau_2(h[n])r_3(h_3[n])T_s - \tau_3(h[n])r_1(h_1[n])T_s$$

The queue size of vehicle 2 at time slot $n+1$ is the following:

$$q_2[n+1] = q_2[n] - \tau_1(h[n])r_2(h_2[n])T_s - \tau_2(h[n])r_3(h_3[n])T_s$$
The linear programming problem for configuration 8 is the following:

$$\begin{align*}
\min_{\tau(h[n])} & \quad U'_{RT,2}(d_2[n])\tau_1(h[n])r_2(h_2[n]) + U'_{RT,2}(d_2[n])\tau_2(h[n])r_3(h_3[n]) \\
& \quad + U'_{RT,1}(d_1[n])\tau_3(h[n])r_1(h_1[n]) \\
\text{s.t} & \quad \tau_1(h[n]) \geq 0 \quad (5.9a) \\
& \quad \tau_2(h[n]) \geq 0 \quad (5.9b) \\
& \quad \tau_3(h[n]) \geq 0 \quad (5.9c) \\
& \quad \tau_1(h[n]) + \tau_2(h[n]) + \tau_3(h[n]) \leq T_s \quad (5.9d) \\
& \quad \tau_1(h[n])r_2(h_2[n]) \leq \frac{q_2[n]}{T_s} \quad (5.9e) \\
& \quad \tau_1(h[n])r_2(h_2[n]) + \tau_2(h[n])r_3(h_3[n]) \leq \frac{q_2[n]}{T_s} \quad (5.9f) \\
& \quad -\tau_2(h[n])r_3(h_3[n]) + \tau_3(h[n])r_1(h_1[n]) \leq \frac{q_1[n]}{T_s} \quad (5.9g) \\
& \quad \tau_2(h[n])r_3(h_3[n]) \leq \frac{q_{\text{max}} - q_1[n]}{T_s}. \quad (5.9h)
\end{align*}$$

The meanings of the constraints are the following:

- (5.9a), (5.9b) and (5.9c) mean that the fractions of time slot must be positive or zero;
- (5.9d) means that the sum of the fractions of time slot must be lower or equal to time slot duration;
- (5.9e), (5.9f) and (5.9g) mean that vehicles cannot transmit more data than those in their queue;
- (5.9h) is a constraint for congestion control that comes from:

$$q_1[n] + \tau_2(h[n])r_3(h_3[n])T_s \leq q_{\text{max}},$$

that means that it is not possible to exceed the maximum queue capacity.

### 5.2.2 Downlink

The downlink scenario is shown in Figure 5.10. We consider only two vehicles, as in uplink, even if it is possible to extend this study to more users. In downlink, we have to consider that the roadside unit maintains separate queues for different users (\(q_1^*[n]\) for vehicle 1 and \(q_2^*[n]\) for vehicle 2). As in uplink, both vehicles have their own queue (vehicle 1 with \(q_1[n]\) and vehicle 2 with \(q_2[n]\)) and there are three different channel gains (\(h_1[n], h_2[n], h_3[n]\)).
Configuration 1

In this configuration the roadside unit transmits directly to the two vehicles, as shown in Figure 5.11.
The queue size of vehicle 1 at time slot \( n + 1 \) is the following:
\[
q_1[n + 1] = q_1[n] + \tau_1(h[n])r_1(h_1[n])T_s
\]
The queue size of vehicle 2 at time slot \( n + 1 \) is the following:
\[
q_2[n + 1] = q_2[n] + \tau_2(h[n])r_2(h_2[n])T_s
\]
The linear programming problem for configuration 1 is the following:
\[
\min_{\tau(h[n])} U'_{RT,1}(d'_1[n])r_1(h[n]) + U'_{RT,2}(d'_2[n])r_2(h_2[n])
\]
s.t.
\[
\tau_1(h[n]) \geq 0 \quad \text{ (5.10a)}
\]
\[
\tau_2(h[n]) \geq 0 \quad \text{ (5.10b)}
\]
\[
\tau_1(h[n]) + \tau_2(h[n]) \leq T_s \quad \text{ (5.10c)}
\]
\[
\tau_1(h[n])r_1(h_1[n]) \leq \frac{q_1[n]}{T_s} \quad \text{ (5.10d)}
\]
\[
\tau_2(h[n])r_2(h_2[n]) \leq \frac{q_2[n]}{T_s} \quad \text{ (5.10e)}
\]
\[
\tau_1(h[n])r_1(h_1[n]) \leq \frac{q_{\text{max}} - q_1[n]}{T_s} \quad \text{ (5.10f)}
\]
\[
\tau_2(h[n])r_2(h_2[n]) \leq \frac{q_{\text{max}} - q_2[n]}{T_s} \quad \text{ (5.10g)}
\]
The meanings of the constraints are the following:

- (5.10a) and (5.10b) mean that the fractions of time slot must be positive or zero;
- (5.10c) means that the sum of the fractions of time slot must be lower or equal to time slot duration;
- (5.10d) and (5.10e) mean that the roadside unit cannot transmit more data than those in its queues;
- (5.10f) and (5.10g) are constraints for congestion control that come from:
\[
q_1[n] + \tau_1(h[n])r_1(h_1[n])T_s \leq q_{\text{max}}
\]
\[
q_2[n] + \tau_2(h[n])r_2(h_2[n])T_s \leq q_{\text{max}},
\]
that mean that it is not possible to exceed the maximum queue capacity.
**Configuration 2**

In this configuration the roadside unit transmits to vehicle 2 and then vehicle 2 transmits to vehicle 1, as shown in Figure 5.12.

The queue size of vehicle 1 at time slot $n + 1$ is the following:

$$q_1[n + 1] = q_1[n] + \tau_2(h[n])r_3(h_3[n])T_s$$

The queue size of vehicle 2 at time slot $n + 1$ is the following:

$$q_2[n + 1] = q_2[n] + \tau_1(h[n])r_2(h_2[n])T_s - \tau_2(h[n])r_3(h_3[n])T_s$$
The linear programming problem for configuration 2 is the following:

$$\min_{\tau(h[n])} U'_{RT,2}(d'_2[n])\tau_1(h[n])r_2(h_2[n]) + U'_{RT,2}(d'_2[n])\tau_2(h[n])r_3(h_3[n])$$

s.t.  
$$\tau_1(h[n]) \geq 0$$  (5.11a)
$$\tau_2(h[n]) \geq 0$$  (5.11b)
$$\tau_1(h[n]) + \tau_2(h[n]) \leq T_s$$  (5.11c)
$$\tau_1(h[n])r_2(h_2[n]) \leq \frac{q_2^*[n]}{T_s}$$  (5.11d)
$$-\tau_1(h[n])r_2(h_2[n]) + \tau_2(h[n])r_3(h_3[n]) \leq \frac{q_2[n]}{T_s}$$  (5.11e)
$$\tau_2(h[n])r_3(h_3[n]) \leq \frac{q_{\text{max}} - q_1[n]}{T_s}$$  (5.11f)
$$\tau_1(h[n])r_2(h_2[n]) \leq \frac{q_{\text{max}} - q_2[n]}{T_s}$$  (5.11g)

The meanings of the constraints are the following:

• (5.11a) and (5.11b) mean that the fractions of time slot must be positive or zero;
• (5.11c) means that the sum of the fractions of time slot must be lower or equal to time slot duration;
• (5.11d) means that the roadside unit cannot transmit more data than those in its queue;
• (5.11e) means that vehicle 2 cannot transmit more data than those in its queue;
• (5.11f) and (5.11g) are constraints for congestion control that come from:
  $$q_1[n] + \tau_2(h[n])r_3(h_3[n])T_s \leq q_{\text{max}}$$
  $$q_2[n] + \tau_1(h[n])r_2(h_2[n])T_s \leq q_{\text{max}},$$
that mean that it is not possible to exceed the maximum queue capacity.
Configuration 3

In this configuration the roadside unit transmits to vehicle 2 for two consecutive fractions of time slot and then vehicle 2 transmits to vehicle 1, as shown in Figure 5.13.

The queue size of vehicle 1 at time slot $n+1$ is the following:

$$q_1[n+1] = q_1[n] + \tau_3(h[n])r_3(h_3[n])T_s$$

The queue size of vehicle 2 at time slot $n+1$ is the following:

$$q_2[n+1] = q_2[n] + \tau_1(h[n])r_2(h_2[n])T_s + \tau_2(h[n])r_2(h_2[n])T_s - \tau_3(h[n])r_3(h_3[n])T_s$$
The linear programming problem for configuration 3 is the following:

\[
\begin{align*}
\min_{\tau(h[n])} & \quad U'_{RT,2}(d_2'[n])\tau_1(h[n])r_2(h_2[n]) + U'_{RT,2}(d_2'[n])\tau_2(h[n])r_2(h_2[n]) \\
& \quad + U'_{RT,2}(d_2'[n])\tau_3(h[n])r_3(h_3[n]) \\
\text{s.to} & \quad \tau_1(h[n]) \geq 0 \quad \text{(5.12a)} \\
& \quad \tau_2(h[n]) \geq 0 \quad \text{(5.12b)} \\
& \quad \tau_3(h[n]) \geq 0 \quad \text{(5.12c)} \\
& \quad \tau_1(h[n]) + \tau_2(h[n]) + \tau_3(h[n]) \leq T_s \quad \text{(5.12d)} \\
& \quad \tau_1(h[n])r_2(h_2[n]) \leq \frac{q_2^*[n]}{T_s} \quad \text{(5.12e)} \\
& \quad \tau_1(h[n])r_2(h_2[n]) + \tau_2(h[n])r_2(h_2[n]) \leq \frac{q_2^*[n]}{T_s} \quad \text{(5.12f)} \\
& \quad -\tau_1(h[n])r_2(h_2[n]) - \tau_2(h[n])r_2(h_2[n]) + \tau_3(h[n])r_3(h_3[n]) \leq \frac{q_2^*[n]}{T_s} \quad \text{(5.12g)} \\
& \quad \tau_3(h[n])r_3(h_3[n]) \leq \frac{q_{max} - q_1[n]}{T_s} \quad \text{(5.12h)} \\
& \quad \tau_1(h[n])r_2(h_2[n]) + \tau_2(h[n])r_2(h_2[n]) \leq \frac{q_{max} - q_2[n]}{T_s}. \quad \text{(5.12i)}
\end{align*}
\]

The meanings of the constraints are the following:

- (5.12a), (5.12b) and (5.12c) mean that the fractions of time slot must be positive or zero;
- (5.12d) means that the sum of the fractions of time slot must be lower or equal to time slot duration;
- (5.12e) and (5.12f) mean that the roadside unit cannot transmit more data than those in its queue;
- (5.12g) means that vehicle 2 cannot transmit more data than those in its queue;
- (5.12h) and (5.12i) are constraints for congestion control that come from:

\[
q_1[n] + \tau_3(h[n])r_3(h_3[n])T_s \leq q_{max} \\
q_2[n] + \tau_1(h[n])r_2(h_2[n])T_s + \tau_2(h[n])r_2(h_2[n])T_s \leq q_{max},
\]

that mean that it is not possible to exceed the maximum queue capacity.
**Configuration 4**

In this configuration the roadside unit transmits directly to the two vehicles and then vehicle 2 transmits to vehicle 1, as shown in Figure 5.14.

![Figure 5.14: Downlink - Configuration 4](image)

The queue size of vehicle 1 at time slot $n + 1$ is the following:

$$ q_1[n + 1] = q_1[n] + \tau_1(h[n])r_1(h_1[n])T_s + \tau_3(h[n])r_3(h_3[n])T_s $$

The queue size of vehicle 2 at time slot $n + 1$ is the following:

$$ q_2[n + 1] = q_2[n] + \tau_2(h[n])r_2(h_2[n])T_s - \tau_3(h[n])r_3(h_3[n])T_s $$
The linear programming problem for configuration 4 is the following:

\[
\begin{align*}
\min_{\tau(h[n])} \quad & U'_{RT,1}(d_1'[n])\tau_1(h[n])r_1(h_1[n]) + U'_{RT,2}(d_2'[n])\tau_2(h[n])r_2(h_2[n]) \\
& + U'_{RT,3}(d_3'[n])\tau_3(h[n])r_3(h_3[n]) \\
\text{s.to} \quad & \tau_1(h[n]) \geq 0 \\
& \tau_2(h[n]) \geq 0 \\
& \tau_3(h[n]) \geq 0 \\
& \tau_1(h[n]) + \tau_2(h[n]) + \tau_3(h[n]) \leq T_s \\
& \tau_1(h[n])r_1(h_1[n]) \leq \frac{q_1[n]}{T_s} \\
& \tau_2(h[n])r_2(h_2[n]) \leq \frac{q_2^*[n]}{T_s} \\
& -\tau_2(h[n])r_2(h_2[n]) + \tau_3(h[n])r_3(h_3[n]) \leq \frac{q_2[n]}{T_s} \\
& \tau_1(h[n])r_1(h_1[n]) + \tau_3(h[n])r_3(h_3[n]) \leq \frac{q_{\max} - q_1[n]}{T_s} \\
& \tau_2(h[n])r_2(h_2[n]) \leq \frac{q_{\max} - q_2[n]}{T_s}. 
\end{align*}
\]

The meanings of the constraints are the following:

- (5.13a), (5.13b) and (5.13c) mean that the fractions of time slot must be positive or zero;
- (5.13d) means that the sum of the fractions of time slot must be lower or equal to time slot duration;
- (5.13e) and (5.13f) mean that the roadside unit cannot transmit more data than those in its queues;
- (5.13g) means that vehicle 2 cannot transmit more data than those in its queue;
- (5.13h) and (5.13i) are constraints for congestion control that come from:

\[
q_1[n] + \tau_1(h[n])r_1(h_1[n])T_s + \tau_3(h[n])r_3(h_3[n])T_s \leq q_{\max}
\]

\[
q_2[n] + \tau_2(h[n])r_2(h_2[n])T_s \leq q_{\max},
\]

that mean that it is not possible to exceed the maximum queue capacity.
Configuration 5

In this case we have four fractions of time slot and the scenario is similar to configuration 4 except that finally the roadside unit transmits again to vehicle 2, as shown in Figure 5.15.

![Figure 5.15: Downlink - Configuration 5](image)

The queue size of vehicle 1 at time slot $n+1$ is the following:

$$q_1[n+1] = q_1[n] + \tau_1(h[n])r_1(h_1[n])T_s + \tau_3(h[n])r_3(h_3[n])T_s$$

The queue size of vehicle 2 at time slot $n+1$ is the following:

$$q_2[n+1] = q_2[n] + \tau_2(h[n])r_2(h_2[n])T_s - \tau_3(h[n])r_3(h_3[n])T_s + \tau_4(h[n])r_2(h_2[n])T_s$$
The linear programming problem for configuration 5 is the following:

\[
\begin{align*}
\min_{\tau(h[n])} & \quad U'_{RT,1}(d'_1[n])\tau_1(h[n])r_1(h_1[n]) + U'_{RT,2}(d'_2[n])\tau_2(h[n])r_2(h_2[n]) \\
& + U'_{RT,2}(d_2[n])\tau_3(h[n])r_3(h_3[n]) + U'_{RT,2}(d'_2[n])\tau_4(h[n])r_2(h_2[n]) \\
\text{s.t.} & \quad \tau_1(h[n]) \geq 0 \quad (5.14a) \\
& \quad \tau_2(h[n]) \geq 0 \quad (5.14b) \\
& \quad \tau_3(h[n]) \geq 0 \quad (5.14c) \\
& \quad \tau_4(h[n]) \geq 0 \quad (5.14d) \\
& \quad \tau_1(h[n]) + \tau_2(h[n]) + \tau_3(h[n]) + \tau_4(h[n]) \leq T_s \quad (5.14e) \\
& \quad \tau_1(h[n])r_1(h_1[n]) \leq \frac{q_1[n]}{T_s} \quad (5.14f) \\
& \quad \tau_2(h[n])r_2(h_2[n]) \leq \frac{q_2[n]}{T_s} \quad (5.14g) \\
& \quad -\tau_2(h[n])r_2(h_2[n]) + \tau_3(h[n])r_3(h_3[n]) \leq \frac{q_2[n]}{T_s} \quad (5.14h) \\
& \quad \tau_2(h[n])r_2(h_2[n]) + \tau_4(h[n])r_2(h_2[n]) \leq \frac{q_2[n]}{T_s} \quad (5.14i) \\
& \quad \tau_1(h[n])r_1(h_1[n]) + \tau_3(h[n])r_3(h_3[n]) \leq \frac{q_{\text{max}} - q_1[n]}{T_s} \quad (5.14j) \\
& \quad \tau_2(h[n])r_2(h_2[n]) - \tau_3(h[n])r_3(h_3[n]) + \tau_4(h[n])r_2(h_2[n]) \leq \frac{q_{\text{max}} - q_2[n]}{T_s} \quad (5.14k) \\
\end{align*}
\]

The meanings of the constraints are the following:

- (5.14a), (5.14b), (5.14c) and (5.14d) mean that the fractions of time slot must be positive or zero;
- (5.14e) means that the sum of the fractions of time slot must be lower or equal to time slot duration;
- (5.14f), (5.14g) and (5.14i) mean that the roadside unit cannot transmit more data than those in its queues;
- (5.14h) means that vehicle 2 cannot transmit more data than those in its queue;
- (5.14j) and (5.14k) are constraints for congestion control that come from:

\[q_1[n] + \tau_1(h[n])r_1(h_1[n])T_s + \tau_3(h[n])r_3(h_3[n])T_s \leq q_{\text{max}}\]
\[
q_2[n] + \tau_2(h[n])r_2(h_2[n])T_s - \tau_3(h[n])r_3(h_3[n])T_s + \tau_4(h[n])r_2(h_2[n])T_s \leq q_{max},
\]
that means that it is not possible to exceed the maximum queue capacity.

**Configuration 6**

In this configuration the roadside unit transmits to vehicle 1 and then vehicle 1 transmits to vehicle 2, as shown in Figure 5.16.

![Figure 5.16: Downlink - Configuration 6](image)

The queue size of vehicle 1 at time slot \( n + 1 \) is the following:

\[
q_1[n + 1] = q_1[n] + \tau_1(h[n])r_1(h_1[n])T_s - \tau_2(h[n])r_3(h_3[n])T_s
\]

The queue size of vehicle 2 at time slot \( n + 1 \) is the following:

\[
q_2[n + 1] = q_2[n] + \tau_2(h[n])r_3(h_3[n])T_s
\]
The linear programming problem for configuration 6 is the following:

\[
\begin{align*}
\min_{\tau(h[n])} & \quad U'_{RT,1}(d_1[n])\tau_1(h[n])r_1(h_1[n]) + U'_{RT,1}(d_1[n])\tau_2(h[n])r_3(h_3[n]) \\
\text{s.to} & \quad \tau_1(h[n]) \geq 0 \quad (5.15a) \\
& \quad \tau_2(h[n]) \geq 0 \quad (5.15b) \\
& \quad \tau_1(h[n]) + \tau_2(h[n]) \leq T_s \quad (5.15c) \\
& \quad \tau_1(h[n])r_1(h_1[n]) \leq \frac{q_1^n}{T_s} \quad (5.15d) \\
& \quad -\tau_1(h[n])r_1(h_1[n]) + \tau_2(h[n])r_3(h_3[n]) \leq \frac{q_1^n}{T_s} \quad (5.15e) \\
& \quad \tau_1(h[n])r_1(h_1[n]) \leq \frac{q_{max} - q_1^n}{T_s} \quad (5.15f) \\
& \quad \tau_2(h[n])r_3(h_3[n]) \leq \frac{q_{max} - q_2^n}{T_s}. \quad (5.15g)
\end{align*}
\]

The meanings of the constraints are the following:

- (5.15a) and (5.15b) mean that the fractions of time slot must be positive or zero;
- (5.15c) means that the sum of the fractions of time slot must be lower or equal to time slot duration;
- (5.15d) means that the roadside unit cannot transmit more data than those in its queue;
- (5.15e) means that vehicle 1 cannot transmit more data than those in its queue;
- (5.15f) and (5.15g) are constraints for congestion control that come from:
  \[
  q_1[n] + \tau_1(h[n])r_1(h_1[n])T_s \leq q_{max} \\
  q_2[n] + \tau_2(h[n])r_3(h_3[n])T_s \leq q_{max},
  \]
  that mean that it is not possible to exceed the maximum queue capacity.
Configuration 7

In this configuration the roadside unit transmits to vehicle 1 for two consecutive fractions of time slot and then vehicle 1 transmits to vehicle 2, as shown in Figure 5.17.

The queue size of vehicle 1 at time slot $n+1$ is the following:

$$q_1[n+1] = q_1[n] + \tau_1(h[n])r_1(h_1[n])T_s + \tau_2(h[n])r_1(h_1[n])T_s - \tau_3(h[n])r_3(h_3[n])T_s$$

The queue size of vehicle 2 at time slot $n+1$ is the following:

$$q_2[n+1] = q_2[n] + \tau_3(h[n])r_3(h_3[n])T_s$$
The linear programming problem for configuration 7 is the following:

\[
\begin{align*}
\text{min} & \quad U'_{RT,1}(d'_{1}[n])\tau_1(h[n])r_1(h_1[n]) + U'_{RT,1}(d'_{1}[n])\tau_2(h[n])r_1(h_1[n]) \\
& \quad + U'_{RT,1}(d'_{1}[n])\tau_3(h[n])r_3(h_3[n]) \\
\text{s.to} & \quad \tau_1(h[n]) \geq 0 \quad (5.16a) \\
& \quad \tau_2(h[n]) \geq 0 \quad (5.16b) \\
& \quad \tau_3(h[n]) \geq 0 \quad (5.16c) \\
& \quad \tau_1(h[n]) + \tau_2(h[n]) + \tau_3(h[n]) \leq T_s \quad (5.16d) \\
& \quad \tau_1(h[n])r_1(h_1[n]) \leq \frac{q_1[n]}{T_s} \quad (5.16e) \\
& \quad \tau_1(h[n])r_1(h_1[n]) + \tau_2(h[n])r_1(h_1[n]) \leq \frac{q_1[n]}{T_s} \quad (5.16f) \\
& \quad -\tau_1(h[n])r_1(h_1[n]) - \tau_2(h[n])r_1(h_1[n]) + \tau_3(h[n])r_3(h_3[n]) \\
& \quad \leq \frac{q_2[n]}{T_s} \quad (5.16g) \\
& \quad \tau_1(h[n])r_1(h_1[n]) + \tau_2(h[n])r_1(h_1[n]) \leq \frac{q_{\text{max}} - q_1[n]}{T_s} \quad (5.16h) \\
& \quad \tau_3(h[n])r_3(h_3[n]) \leq \frac{q_{\text{max}} - q_2[n]}{T_s} \quad (5.16i)
\end{align*}
\]

The meanings of the constraints are the following:

- (5.16a), (5.16b) and (5.16c) mean that the fractions of time slot must be positive or zero;
- (5.16d) means that the sum of the fractions of time slot must be lower or equal to time slot duration;
- (5.16e) and (5.16f) mean that the roadside unit cannot transmit more data than those in its queue;
- (5.16g) means that vehicle 1 cannot transmit more data than those in its queue;
- (5.16h) and (5.16i) are constraints for congestion control that come from:

\[
q_1[n] + \tau_1(h[n])r_1(h_1[n])T_s + \tau_2(h[n])r_1(h_1[n])T_s \leq q_{\text{max}} \\
q_2[n] + \tau_3(h[n])r_3(h_3[n])T_s \leq q_{\text{max}},
\]

that mean that it is not possible to exceed the maximum queue capacity.
Configuration 8

In this configuration the roadside unit transmits directly to the two vehicles and then vehicle 1 transmits to vehicle 2, as shown in Figure 5.18.

The queue size of vehicle 1 at time slot $n + 1$ is the following:

$$q_1[n + 1] = q_1[n] + \tau_2(h[n])r_1(h_1[n])T_s - \tau_3(h[n])r_3(h_3[n])T_s$$

The queue size of vehicle 2 at time slot $n + 1$ is the following:

$$q_2[n + 1] = q_2[n] + \tau_1(h[n])r_2(h_2[n])T_s + \tau_3(h[n])r_3(h_3[n])T_s$$
The linear programming problem for configuration 8 is the following:

\[
\begin{align*}
\min_{\tau(h[n])} & \quad U'_{RT,2}(d'_2[n])\tau_1(h[n])r_2(h_2[n]) + U'_{RT,1}(d'_1[n])\tau_2(h[n])r_1(h_1[n]) \\
& + U'_{RT,1}(d_1[n])\tau_3(h[n])r_3(h_3[n]) \\
\text{s.to} \quad & \tau_1(h[n]) \geq 0 \quad (5.17a) \\
& \tau_2(h[n]) \geq 0 \quad (5.17b) \\
& \tau_3(h[n]) \geq 0 \quad (5.17c) \\
& \tau_1(h[n]) + \tau_2(h[n]) + \tau_3(h[n]) \leq T_s \quad (5.17d) \\
& \tau_1(h[n])r_2(h_2[n]) \leq \frac{q^*_2[n]}{T_s} \quad (5.17e) \\
& \tau_2(h[n])r_1(h_1[n]) \leq \frac{q^*_1[n]}{T_s} \quad (5.17f) \\
& -\tau_2(h[n])r_1(h_1[n]) + \tau_3(h[n])r_3(h_3[n]) \leq \frac{q_1[n]}{T_s} \quad (5.17g) \\
& \tau_2(h[n])r_1(h_1[n]) \leq \frac{q_{\max} - q_1[n]}{T_s} \quad (5.17h) \\
& \tau_1(h[n])r_2(h_2[n]) + \tau_3(h[n])r_3(h_3[n]) \leq \frac{q_{\max} - q_2[n]}{T_s}. \quad (5.17i)
\end{align*}
\]

The meanings of the constraints are the following:

- (5.17a), (5.17b) and (5.17c) mean that the fractions of time slot must be positive or zero;
- (5.17d) means that the sum of the fractions of time slot must be lower or equal to time slot duration;
- (5.17e) and (5.17f) mean that the roadside unit cannot transmit more data than those in its queues;
- (5.17g) means that vehicle 1 cannot transmit more data than those in its queue;
- (5.17h) and (5.17i) are constraints for congestion control that come from:
  \[
  q_1[n] + \tau_2(h[n])r_1(h_1[n])T_s \leq q_{\max} \\
  q_2[n] + \tau_1(h[n])r_2(h_2[n])T_s + \tau_3(h[n])r_3(h_3[n])T_s \leq q_{\max},
  \]
  that means that it is not possible to exceed the maximum queue capacity.

In the next chapter we propose some results obtained by the simulations of all these linear programming problems.
Chapter 6

Simulation results

In this chapter we report the results obtained by Matlab simulations by using the function linprog. We have done simulations to solve all the linear programming problems presented in Chapter 5.

Three different channel gain scenarios have been considered, using the following numerical values chosen according to IEEE 802.11p:

- $U_{RT,j}(d_j[n]) = -d_j[n]^2$, for $d_j[n] > 0$;
- $U'_{RT,j}(d_j[n]) = -2d_j[n]$, for $d_j[n] > 0$;
- $T_s = 100$ ms;
- $P = 33$ dBm = 1995 mW;
- $h_i[n] = 0.1 \Rightarrow r_i(h_i[n]) = 7.65$ bit/s;
- $h_i[n] = 0.4 \Rightarrow r_i(h_i[n]) = 9.64$ bit/s;
- $h_i[n] = 0.5 \Rightarrow r_i(h_i[n]) = 9.96$ bit/s;
- $q_1[n] = q_2[n] = q_1^*[n] = q_2^*[n] = 50$ kbit;
- $d_1[n] = d_2[n] = d_1^*[n] = d_2^*[n] = 5$ ms;
- $q_{\text{max}} = 500$ kbit.

6.1 Scenario 1

In this scenario we consider $h_1[n] = 0.1$, $h_2[n] = 0.1$ and $h_3[n] = 0.5$, as shown in Figure 6.1.

The results obtained for the uplink are the following:
Figure 6.1: Scenario 1

- Configuration 1:
  \[ \tau_1(h[n]) = 50 \text{ ms} \]
  \[ \tau_2(h[n]) = 50 \text{ ms} \]
  \[ v = -7.6500e + 003 \]

- Configuration 2:
  \[ \tau_1(h[n]) = 50.2008 \text{ ms} \]
  \[ \tau_2(h[n]) = 49.7992 \text{ ms} \]
  \[ v = -8.8096e + 003 \]

- Configuration 3:
  \[ \tau_1(h[n]) = 4.674 \text{ ms} \]
  \[ \tau_2(h[n]) = 50.2008 \text{ ms} \]
  \[ \tau_3(h[n]) = 45.1252 \text{ ms} \]
  \[ v = -8.8096e + 003 \]

- Configuration 4:
  \[ \tau_1(h[n]) = 0 \text{ ms} \]
  \[ \tau_2(h[n]) = 50.2008 \text{ ms} \]
  \[ \tau_3(h[n]) = 49.7992 \text{ ms} \]
  \[ v = -8.8096e + 003 \]
• Configuration 5:
  \[ \tau_1(h[n]) = 0 \text{ ms} \]
  \[ \tau_2(h[n]) = 50.2008 \text{ ms} \]
  \[ \tau_3(h[n]) = 15.7358 \text{ ms} \]
  \[ \tau_4(h[n]) = 34.0634 \text{ ms} \]
  \[ v = -8.8096e + 003 \]

• Configuration 6:
  \[ \tau_1(h[n]) = 50.2008 \text{ ms} \]
  \[ \tau_2(h[n]) = 49.7992 \text{ ms} \]
  \[ v = -8.8096e + 003 \]

• Configuration 7:
  \[ \tau_1(h[n]) = 4.674 \text{ ms} \]
  \[ \tau_2(h[n]) = 50.2008 \text{ ms} \]
  \[ \tau_3(h[n]) = 45.1252 \text{ ms} \]
  \[ v = -8.8096e + 003 \]

• Configuration 8:
  \[ \tau_1(h[n]) = 0 \text{ ms} \]
  \[ \tau_2(h[n]) = 50.2008 \text{ ms} \]
  \[ \tau_3(h[n]) = 49.7992 \text{ ms} \]
  \[ v = -8.8096e + 003 \]

We can see that all the configurations have the same value \( v \) lower than the value of configuration 1, so it is better to choose any configuration except configuration 1.

The results obtained for the downlink are the following:

• Configuration 1:
  \[ \tau_1(h[n]) = 50 \text{ ms} \]
  \[ \tau_2(h[n]) = 50 \text{ ms} \]
  \[ v = -7.6500e + 003 \]
• Configuration 2:
\[ \tau_1(h[n]) = 28.1658 \text{ ms} \]
\[ \tau_2(h[n]) = 71.8342 \text{ ms} \]
\[ v = -9.3094e+003 \]

• Configuration 3:
\[ \tau_1(h[n]) = 18.2364 \text{ ms} \]
\[ \tau_2(h[n]) = 9.9294 \text{ ms} \]
\[ \tau_3(h[n]) = 71.8342 \text{ ms} \]
\[ v = -9.3094e+003 \]

• Configuration 4:
\[ \tau_1(h[n]) = 0 \text{ ms} \]
\[ \tau_2(h[n]) = 28.1658 \text{ ms} \]
\[ \tau_3(h[n]) = 71.8342 \text{ ms} \]
\[ v = -9.3094e+003 \]

• Configuration 5:
\[ \tau_1(h[n]) = 0 \text{ ms} \]
\[ \tau_2(h[n]) = 28.1658 \text{ ms} \]
\[ \tau_3(h[n]) = 71.8342 \text{ ms} \]
\[ \tau_4(h[n]) = 0 \text{ ms} \]
\[ v = -9.3094e+003 \]

• Configuration 6:
\[ \tau_1(h[n]) = 28.1658 \text{ ms} \]
\[ \tau_2(h[n]) = 71.8342 \text{ ms} \]
\[ v = -9.3094e+003 \]

• Configuration 7:
\[ \tau_1(h[n]) = 18.2364 \text{ ms} \]
\[ \tau_2(h[n]) = 9.9294 \text{ ms} \]
\[ \tau_3(h[n]) = 71.8342 \text{ ms} \]
\[ v = -9.3094e+003 \]
• Configuration 8:

\[ \tau_1(h[n]) = 0 \text{ ms} \]
\[ \tau_2(h[n]) = 28.1658 \text{ ms} \]
\[ \tau_3(h[n]) = 71.8342 \text{ ms} \]
\[ v = -9.3094e + 003 \]

We can see that all the configurations have the same value \( v \) lower than the value of configuration 1, so it is better to choose any configuration except configuration 1.

6.2 Scenario 2

In this scenario we consider \( h_1[n] = 0.4 \), \( h_2[n] = 0.1 \) and \( h_3[n] = 0.5 \), as shown in Figure 6.2.

![Figure 6.2: Scenario 2](image)

The results obtained for the uplink are the following:

• Configuration 1:

\[ \tau_1(h[n]) = 51.8672 \text{ ms} \]
\[ \tau_2(h[n]) = 48.1328 \text{ ms} \]
\[ v = -8.6822e + 003 \]
- Configuration 2: 
  \[ \tau_1(h[n]) = 50.2008 \text{ ms} \]
  \[ \tau_2(h[n]) = 49.7992 \text{ ms} \]
  \[ v = -8.8096e + 003 \]

- Configuration 3: 
  \[ \tau_1(h[n]) = 4.674 \text{ ms} \]
  \[ \tau_2(h[n]) = 50.2008 \text{ ms} \]
  \[ \tau_3(h[n]) = 45.1252 \text{ ms} \]
  \[ v = -8.8096e + 003 \]

- Configuration 4: 
  \[ \tau_1(h[n]) = 0 \text{ ms} \]
  \[ \tau_2(h[n]) = 50.2008 \text{ ms} \]
  \[ \tau_3(h[n]) = 49.7992 \text{ ms} \]
  \[ v = -8.8096e + 003 \]

- Configuration 5: 
  \[ \tau_1(h[n]) = 0 \text{ ms} \]
  \[ \tau_2(h[n]) = 50.2008 \text{ ms} \]
  \[ \tau_3(h[n]) = 8.0797 \text{ ms} \]
  \[ \tau_4(h[n]) = 41.7195 \text{ ms} \]
  \[ v = -8.8096e + 003 \]

- Configuration 6: 
  \[ \tau_1(h[n]) = 50.2008 \text{ ms} \]
  \[ \tau_2(h[n]) = 49.7992 \text{ ms} \]
  \[ v = -9.8006e + 003 \]

- Configuration 7: 
  \[ \tau_1(h[n]) = 5.1394 \text{ ms} \]
  \[ \tau_2(h[n]) = 50.2008 \text{ ms} \]
  \[ \tau_3(h[n]) = 44.6598 \text{ ms} \]
  \[ v = -9.8006e + 003 \]
• Configuration 8:

$$\tau_1(h[n]) = 0 \text{ ms}$$
$$\tau_2(h[n]) = 50.2008 \text{ ms}$$
$$\tau_3(h[n]) = 49.7992 \text{ ms}$$
$$v = -9.8006e + 003$$

We can see that configuration 6, configuration 7 and configuration 8 have the same value v lower than the values of the other configurations, so it is better to choose among these three configurations.

The results obtained for the downlink are the following:

• Configuration 1:

$$\tau_1(h[n]) = 51.8672 \text{ ms}$$
$$\tau_2(h[n]) = 48.1328 \text{ ms}$$
$$v = -8.6822e + 003$$

• Configuration 2:

$$\tau_1(h[n]) = 28.1658 \text{ ms}$$
$$\tau_2(h[n]) = 71.8342 \text{ ms}$$
$$v = -9.3094e + 003$$

• Configuration 3:

$$\tau_1(h[n]) = 18.2364 \text{ ms}$$
$$\tau_2(h[n]) = 9.9294 \text{ ms}$$
$$\tau_3(h[n]) = 71.8342 \text{ ms}$$
$$v = -9.3094e + 003$$

• Configuration 4:

$$\tau_1(h[n]) = 49.7992 \text{ ms}$$
$$\tau_2(h[n]) = 0 \text{ ms}$$
$$\tau_3(h[n]) = 50.2008 \text{ ms}$$
$$v = -9.8006e + 003$$
- Configuration 5:
  \[
  \begin{align*}
  \tau_1(h[n]) &= 49.7992 \text{ ms} \\
  \tau_2(h[n]) &= 0 \text{ ms} \\
  \tau_3(h[n]) &= 50.2008 \text{ ms} \\
  \tau_4(h[n]) &= 0 \text{ ms} \\
  v &= -9.8006e + 003 
  \end{align*}
  \]

- Configuration 6:
  \[
  \begin{align*}
  \tau_1(h[n]) &= 25.3061 \text{ ms} \\
  \tau_2(h[n]) &= 74.6939 \text{ ms} \\
  v &= -9.8790e + 003 
  \end{align*}
  \]

- Configuration 7:
  \[
  \begin{align*}
  \tau_1(h[n]) &= 15.0397 \text{ ms} \\
  \tau_2(h[n]) &= 10.2664 \text{ ms} \\
  \tau_3(h[n]) &= 74.6939 \text{ ms} \\
  v &= -9.8790e + 003 
  \end{align*}
  \]

- Configuration 8:
  \[
  \begin{align*}
  \tau_1(h[n]) &= 0 \text{ ms} \\
  \tau_2(h[n]) &= 25.3061 \text{ ms} \\
  \tau_3(h[n]) &= 74.6939 \text{ ms} \\
  v &= -9.8790e + 003 
  \end{align*}
  \]

We can see that configuration 6, configuration 7 and configuration 8 have the same value \( v \) lower than the values of the other configurations, so it is better to choose among these three configurations.
6.3 Scenario 3

In this scenario we consider \( h_1[n] = 0.5 \), \( h_2[n] = 0.4 \) and \( h_3[n] = 0.1 \), as shown in Figure 6.3.

The results obtained for the uplink are the following:

- **Configuration 1:**
  \[
  \tau_1(h[n]) = 50.2008 \text{ ms} \\
  \tau_2(h[n]) = 49.7992 \text{ ms} \\
  v = -9.8006e + 003
  \]

- **Configuration 2:**
  \[
  \tau_1(h[n]) = 26.8363 \text{ ms} \\
  \tau_2(h[n]) = 73.1637 \text{ ms} \\
  v = -9.1060e + 003
  \]

- **Configuration 3:**
  \[
  \tau_1(h[n]) = 3.8481 \text{ ms} \\
  \tau_2(h[n]) = 26.8363 \text{ ms} \\
  \tau_3(h[n]) = 69.3156 \text{ ms} \\
  v = -9.1060e + 003
  \]
• Configuration 4:
  \[
  \tau_1(h[n]) = 50.2008 \text{ ms} \\
  \tau_2(h[n]) = 0 \text{ ms} \\
  \tau_3(h[n]) = 49.7992 \text{ ms} \\
  v = -9.8006e + 003
  \]

• Configuration 5:
  \[
  \tau_1(h[n]) = 50.2008 \text{ ms} \\
  \tau_2(h[n]) = 0 \text{ ms} \\
  \tau_3(h[n]) = 10.9478 \text{ ms} \\
  \tau_4(h[n]) = 38.8514 \text{ ms} \\
  v = -9.8006e + 003
  \]

• Configuration 6:
  \[
  \tau_1(h[n]) = 28.1658 \text{ ms} \\
  \tau_2(h[n]) = 71.8342 \text{ ms} \\
  v = -9.3094e + 003
  \]

• Configuration 7:
  \[
  \tau_1(h[n]) = 3.7415 \text{ ms} \\
  \tau_2(h[n]) = 28.1658 \text{ ms} \\
  \tau_3(h[n]) = 68.0927 \text{ ms} \\
  v = -9.3094e + 003
  \]

• Configuration 8:
  \[
  \tau_1(h[n]) = 49.7992 \text{ ms} \\
  \tau_2(h[n]) = 0 \text{ ms} \\
  \tau_3(h[n]) = 50.2008 \text{ ms} \\
  v = -9.8006e + 003
  \]

We can see that configuration 1, configuration 4, configuration 5 and configuration 8 have the same value v lower than the values of the other configurations, so it is better to choose among these four configurations.

The results obtained for the downlink are the following:
• Configuration 1: 
\[
\begin{align*}
\tau_1(h[n]) &= 50.2008 \text{ ms} \\
\tau_2(h[n]) &= 49.7992 \text{ ms} \\
v &= -9.8006e + 003
\end{align*}
\]

• Configuration 2: 
\[
\begin{align*}
\tau_1(h[n]) &= 51.8672 \text{ ms} \\
\tau_2(h[n]) &= 48.1328 \text{ ms} \\
v &= -8.6822e + 003
\end{align*}
\]

• Configuration 3: 
\[
\begin{align*}
\tau_1(h[n]) &= 27.5110 \text{ ms} \\
\tau_2(h[n]) &= 24.3562 \text{ ms} \\
\tau_3(h[n]) &= 48.1328 \text{ ms} \\
v &= -8.6822e + 003
\end{align*}
\]

• Configuration 4: 
\[
\begin{align*}
\tau_1(h[n]) &= 50.2008 \text{ ms} \\
\tau_2(h[n]) &= 49.7992 \text{ ms} \\
\tau_3(h[n]) &= 0 \text{ ms} \\
v &= -9.8006e + 003
\end{align*}
\]

• Configuration 5: 
\[
\begin{align*}
\tau_1(h[n]) &= 50.2008 \text{ ms} \\
\tau_2(h[n]) &= 13.2453 \text{ ms} \\
\tau_3(h[n]) &= 0 \text{ ms} \\
\tau_4(h[n]) &= 36.5539 \text{ ms} \\
v &= -9.8006e + 003
\end{align*}
\]

• Configuration 6: 
\[
\begin{align*}
\tau_1(h[n]) &= 50.2008 \text{ ms} \\
\tau_2(h[n]) &= 49.7992 \text{ ms} \\
v &= -8.8096e + 003
\end{align*}
\]
• Configuration 7:
  \[ \tau_1(h[n]) = 27.357 \text{ ms} \]
  \[ \tau_2(h[n]) = 22.8438 \text{ ms} \]
  \[ \tau_3(h[n]) = 49.7992 \text{ ms} \]
  \[ v = -8.8096 e + 003 \]

• Configuration 8:
  \[ \tau_1(h[n]) = 49.7992 \text{ ms} \]
  \[ \tau_2(h[n]) = 50.2008 \text{ ms} \]
  \[ \tau_3(h[n]) = 0 \text{ ms} \]
  \[ v = -9.8006 e + 003 \]

We can see that configuration 1, configuration 4, configuration 5 and configuration 8 have the same value of lower than the values of the other configurations, so it is better to choose among these four configurations.

As it is possible to see in the results obtained by all the simulations, the duration of the fraction of time slot depends on the channel conditions, so a higher channel gain corresponds to a higher transmission time whereas a lower channel gain corresponds to a lower transmission time. It is also possible to see that the case where vehicles transmit directly to the roadside unit in uplink is not always the best choice because of the channel conditions. In the same way, the configuration where the roadside unit transmits directly to the vehicles in downlink is not always the best choice.
Chapter 7

Conclusions and future works

In this thesis we introduced a new real-time scheduling algorithm both for uplink and downlink communications, suitable for ITS applications. It was based on a TDMA MAC method, where the roadside unit had the tasks to estimate the channel conditions and assign fractions of time slot to users. As it was possible to see in the results obtained by the Matlab simulations, the duration of the fraction of time slot depended on the channel conditions, so a higher channel gain corresponded to a higher transmission time whereas a lower channel gain corresponded to a lower transmission time. As it was also possible to see by the simulations of the different configurations involved in the algorithm, the case where vehicles transmitted directly to the roadside unit in uplink was not always the best choice because of the channel conditions. In the same way, the configuration where the roadside unit transmitted directly to the vehicles in downlink was not always the best choice. We also introduced a constraint that avoided the congestion of the queues of the vehicles, so no packet was dropped.

The possible future works will be the following:

• Make a comparison between the proposed scheduling algorithm and other existing algorithms;

• Increase the number of vehicles;

• Try to obtain a generalized linear programming problem containing all the considered configurations by using binary variables;

• Use branch and bound method to discard some configurations.
Bibliography


