



HAL
open science

Realistic driver behavior modeling for energy efficiency estimation during heavy-trucks vehicles development

Claire D Agostino

► **To cite this version:**

Claire D Agostino. Realistic driver behavior modeling for energy efficiency estimation during heavy-trucks vehicles development. Other. Ecole Centrale de Lyon, 2014. English. NNT : 2014ECDL0034 . tel-04961503

HAL Id: tel-04961503

<https://theses.hal.science/tel-04961503v1>

Submitted on 21 Feb 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

N° ordre: 2014-34

THÈSE DE L'UNIVERSITÉ DE LYON

Delivrée par

L'ÉCOLE CENTRALE LYON

En vue d'obtenir le titre de

DOCTEUR

SPÉCIALITÉ: INFORMATIQUE

ÉCOLE DOCTORALE INFORMATIQUE ET MATHÉMATIQUES

LABORATOIRE D'INFORMATIQUE EN IMAGE ET SYSTÈMES D'INFORMATION

Modélisation de comportements de conducteurs réalistes pour l'estimation de l'efficacité énergétique durant le développement des véhicules poids-lourds

Soutenue le 27 novembre 2014 par

Mlle Claire D'AGOSTINO

Devant le jury composé de:

<i>Rapporteurs:</i>	Jean-Christophe POPIEUL	Professeur, Université de Valenciennes
	Mohamed HAMMAMI	Maitre de conférences - HDR, Université de Sfax
<i>Directeur:</i>	Liming CHEN	Professeur, Ecole Centrale Lyon
<i>Co-directeur:</i>	Alexandre SAIDI	Maitre de conférences, Ecole Centrale Lyon
<i>Président du jury:</i>	Catherine GARBAY	Directeur de recherches, CNRS
<i>Examineurs:</i>	Stéphane ESPIÉ	Directeur de recherches, IFSTTAR
	Gilles SCOUARNEC	Encadrant, VOLVO



N°:2014-34

UNIVERSITÉ DE LYON THESIS

Delivered by

ÉCOLE CENTRALE LYON

Leading to

DOCTORATE DEGREE

SPECIALITY: COMPUTER SCIENCE

ÉCOLE DOCTORALE INFORMATIQUE ET MATHÉMATIQUES

LABORATOIRE D'INFORMATIQUE EN IMAGE ET SYSTÈMES D'INFORMATION

Realistic driver behavior modeling for energy efficiency estimation during heavy-trucks vehicles development

Submitted and defended on November, 27th 2014, by

Claire D'AGOSTINO

Jury:

<i>Rapporteurs:</i>	Jean-Christophe POPIEUL	Professor, University of Valenciennes
	Mohamed HAMMAMI	Lecturer - HDR, University of Sfax
<i>Director:</i>	Liming CHEN	Professor, Ecole Centrale Lyon
<i>Co-director:</i>	Alexandre SAIDI	Lecturer, Ecole Centrale Lyon
<i>President:</i>	Catherine GARBAY	Research Director, CNRS
<i>Examiners:</i>	Stéphane ESPIÉ	Research Director, IFSTTAR
	Gilles SCOUARNEC	Industrial supervisor, VOLVO





VOLVO

Cette thèse et les informations qu'elle contient sont confidentielles. Elles ne doivent être ni reproduites, ni communiquées, ni diffusées sans l'autorisation préalable écrite de **AB VOLVO**.

Le résumé, la table des matières et le résumé de chaque chapitre en français sont disponibles à la fin de ce rapport.

This PhD report and its content are confidential. This report or its content may not be reproduced, transferred, distributed or stored without prior written permission by **AB Volvo**.

Abstract, table of contents, and summaries of each section, are available in French at the end of the report.

*Success is not final, failure is not fatal:
it is the courage to continue that counts.*
Winston S. Churchill

Acknowledgments

Many thanks to all those who have followed this work, directly or indirectly, who have supported me, and who I have met during these three years thanks to this thesis.

First of all I would like to thank Gilles SCOUARNEC and Pierre LAVORINI for their trust. Thank you for letting me join your team, thank you for your support, and thank you for your advice. I also thank my director and co-director Prof. Liming CHEN and Dr. Alexandre SAIDI for guiding me and for showing me what a research work was.

My thanks go also to the jury members, Jean-Christophe POPIEUL and Mohamed HAMMAMI as rapporteurs, Catherine GARBAY and Stéphane ESPIE. I am proud that they all accepted to spend some time reading and assessing my work.

Special thanks to all the *Driving Performance and Transport Effectiveness* team for the way they shared information professionally, without forgetting the friendly exchanges. I have sincerely learnt a lot thanks to all of you. Thank you to the students who worked with me, Cindi, Mathieu, and Thomas. More generally, a big thank you to all the other teams and colleagues who were involved at different levels, especially the *Test and Simulation support* team and Cédric. Thank you all the people, engineers, technicians, drivers, suppliers, temporary workers, PhD students (Baptiste, Thibault), who helped me to achieve this work. They all shared their experience with me and allowed me to move forward professionally.

Thank you Dad. Finally we are not able to graduate together at the same time, but let's say that my graduation will be your graduation too.

And of course I would like to express my greatest gratitude to my friends and my family.

Contents

1	Introduction	1
1.1	Context and motivations of the study	1
1.1.1	Fuel economy context	1
1.1.2	Road freight context: a complex challenge	2
1.1.3	A specificity of the truck context: rational driving vs. eco-driving	4
1.2	Benefits of modeling during the development of vehicles	5
1.2.1	Test facilities for modeling	6
1.3	Objectives and issues	10
1.4	Approach and contributions	11
1.4.1	Approach	11
1.4.2	Contributions	12
1.5	Thesis Organization	13
1.6	List of Publications	14
2	Knowledge extraction from driving data and driver models	17
2.1	Knowledge Discovery in Databases (KDD) process	17
2.2	Knowledge discovery in driving data	19
2.2.1	Driving data	19
2.2.2	Driving events definition	20
2.2.3	Definition of driving styles	22
2.2.4	Driving features	25
2.2.5	Machine-learning methods and results	29
2.2.6	The selected learning-based algorithms	32
2.2.7	Summary on knowledge discovery with driving data	39
2.3	Related work on driver models	39
2.3.1	Historical background	39
2.3.2	Adapt a driver model to different driving behaviors	41
3	Driving events recognition	51
3.1	Driving events definition	52
3.1.1	Throttle=0 events	52
3.1.2	Throttle>0 events	54
3.2	Driving data acquisition	56
3.2.1	Data collection for the recognition of Throttle=0 events	56
3.2.2	Data collection for previous work validation and for the recognition of Throttle>0 events	57
3.3	Driving events sensitive features	59
3.3.1	Throttle=0 events	61
3.3.2	Throttle>0 events	61

3.4	Driving events recognition from real driving data using two machine-learning algorithms	62
3.4.1	Classification of throttle=0 events	62
3.4.2	Impact of vehicle and driver on Throttle=0 events recognition	66
3.4.3	Throttle>0 events classification	66
3.4.4	Summary of events recognition from real driving data	67
3.5	Driving events recognition in digital roads	68
3.5.1	Digital roads definition	68
3.5.2	Database	69
3.5.3	Results	69
3.6	Conclusion	70
4	Correlation of driver behavior parameters with RDI	73
4.1	Rational driving index	74
4.2	Driving data collection	75
4.2.1	Vehicle	75
4.2.2	Drivers	75
4.2.3	Road	78
4.2.4	Differences in fuel consumption and average speed between drivers	78
4.3	Hierarchical classification	81
4.4	Rational driving-related driving features	83
4.4.1	On extra-urban road	83
4.4.2	On highway	86
4.5	Simulation-based impact study of driving features when crossing a roundabout	89
4.5.1	Selection of a representative roundabout and its representation on the simulation tool	89
4.5.2	Simulating driver behaviors on the digital roundabout	90
4.5.3	Impact of the simulated driving features on RDI	92
4.6	Simulation-based impact study of driving features when approaching a toll	94
4.6.1	Simulating driver behaviors on the simulated toll	95
4.6.2	Impact of the percentage of coasting on RDI	95
4.7	RDI correlation to a cost model	96
4.8	Conclusion	99
5	Driver model generator	101
5.1	Cycle Tuning Structure	101
5.1.1	Driving events identification	104
5.1.2	Driver efficiency energy management (EEM) and vehicle dynamic identification	105
5.1.3	Cycle tuning	105
5.1.4	Self-learning cycle	107

5.1.5	Chassis-dynamometer and simulation controllers	110
5.2	Implementation	110
5.3	Results	112
5.3.1	Roundabout	112
5.3.2	Toll	112
5.3.3	Stops	115
5.3.4	Combination of several events	117
5.3.5	Summary	119
6	Conclusion	121
A	Truck usages	125
B	List of driving events	127
C	Driving features in related work	129
D	Data collection documents	139
D.1	Driver questionnaire	139
D.2	Accompanying person questionnaire	144
	Bibliography	147

List of Figures

1.1	Breakdown of freight in the EU. 78% of the freight is transported by the road in 2010.	1
1.2	Proportion of transportation (%) in the total energetic consumption of EU	2
1.3	Structure of vehicle cost price in 2013. Source: Enquete CNR Longue Distance 2013	2
1.4	Rigid on the left, Tractor on the middle, Rigid and trailer on the right.	3
1.5	All the test means during the truck development process. GSP is our simulation tool presented in the following section.	7
1.6	Screenshot of GSP simulation tool made with simulink models. The three main components can be seen: road, driver and vehicle.	7
1.7	Testing a truck on the chassis-dynamometer	8
1.8	Main components of a test on chassis-dynamometer. Arrows represent data exchanges between them.	8
1.9	The main simulation parts in the simulation tool. Arrows represent exchanges of data between driver, vehicle and road.	9
1.10	Overall approach of our work.	12
1.11	Model learning using data mining. We can see the output format of two machine learning algorithms used in this study: on the left the decision tree, and the linear logistic regression on the right. As it can be seen, we can easily interpret these models as they keep the physical sense of the features.	13
2.1	An Overview of the Steps That Compose the KDD Process	18
2.2	Explanations of algorithms utilization. We can see the output form of the two algorithms: on the left the decision tree, and on the right linear logistic regression. These models are observed to be easily interpretable as they retain the physical sense of the driving features.	32
2.3	Example of visualization of a part of a decision tree.	35
2.4	Example of visualization of a part of the LLR output model.	37
2.5	Simulation results for three driver types with Kiencke and Nielsen driver models.	43
2.6	IPG driver parameters for aggressive (on the left) and normal (on the right) drivers.	45
2.7	GSP driver behavior (top graph) and IPG driver behavior (bottom graph) during a braking phase.	46
2.8	General architecture of COSMODRIVE [1]	46
2.9	An example of COSMODRIVE driving schema: left turn at an urban crossroads [2]	47

2.10	Example of acceleration strategy implementation in the ACEA driver model.	49
3.1	Actuators that the driver can use to control the speed of the truck. .	53
3.2	Illustration of the two types of driving events: a <i>throttle>0</i> event followed by a <i>throttle=0</i> event. In the first case, throttle pedal is in use whereas brake pedal is engaged in the second case	53
3.3	Examples of driving events: speed-bump, red traffic light, curve, give way, roundabout, lower speed limit, truck size constraint (2 examples), keep constant speed in downhill, traffic, toll, access ramp, stop and low speed, respectively.	54
3.4	FL vehicle	57
3.5	This figure represents the stages from the data collection to the driving events. A data logger acquires signals from CAN bus, GPS and camera. Once all the data is collected, we segment the driving data into driving events. After that, we can calculate the features values, and use the database to perform the classification.	57
3.6	RENAULT TRUCKS Premium	58
3.7	Box events	58
3.8	Illustration of the Fisher distance between classes. On the left, distance to toll class. On the right, distance to traffic class.	64
3.9	Example of a digital road: <i>a</i> is the square signal of speed vs. distance. On graph <i>b</i> each point represents a stop. <i>c</i> is the slope vs. distance function.	69
3.10	Example of identified events on a digital road. Letters represent the different events.	70
4.1	The Renault Trucks D-wide vehicle, a heavy multi-purpose vehicle, used to acquire real driving data. This vehicle is quite representative of vehicles used for regional distribution.	76
4.2	Camera for video-recording used during the data collection.	76
4.3	Years of driving experience of drivers <i>vs.</i> <i>RDI</i>	77
4.4	Age of drivers <i>vs.</i> <i>RDI</i>	77
4.5	Interests of the drivers. They gave their answer on a scale from 1 to 10 for each subject (10=highest mark).	78
4.6	Road of the data collection. Framed roads are noticeable sections of the path.	79
4.7	Gaps with average <i>fc</i> (blue) and average speed (red) for each driver (real lap only). <i>RDI</i> is represented in green. The best drivers are on the right.	80
4.8	Distribution of <i>RDI</i> for the 34 drivers of the data collection compared to the distribution of <i>RDI</i> from customer data: 939 Midlum and 2496 Premium D.	80
4.9	First method for driver classification.	82

4.10	Hierarchical method for driver classification.	82
4.11	Classification rate of the driver type classification with C4.5 and LLR algorithms	82
4.12	The extra-urban section which includes 3 roundabouts.	83
4.13	Distribution of the <i>RDI</i> s of the drivers on the 75 runs. We separated drivers in the three classes with the help of the range defined as: $[(\mu_{RDI} - \sigma_{RDI}); (\mu_{RDI} + \sigma_{RDI})]$ where μ_{RDI} is the average value and σ_{RDI} is the standard deviation of the <i>RDI</i> s.	84
4.14	Distribution of <i>RDI</i> s on Highway section.	87
4.15	View of the selected roundabout and the driving direction.	90
4.16	Selected roundabout model for simulation. a) is the speed target vs. distance. On b), the black point represents the final stop. c) is the road-type. And d) is the slope vs. distance signal.	91
4.17	The main simulation parts in the simulation tool. Arrows represent exchange of data between driver, vehicle and road.	92
4.18	Speed vs. distance signals from the simulations for the five selected runs. We can observe the different cruising speeds, crossing speeds (at 1810m) and the deceleration values.	93
4.19	Google earth view of the toll	95
4.20	Digital representation of the highway section. At the top, there is the speed target at 90 km/h vs. distance signal. In the middle, the black points represent the stops. At the bottom, this is the signal of the slope vs. distance.	95
4.21	Digital representation of the highway section	96
4.22	FT (€ per ton.km) vs. <i>RDI</i> values for different runs. For higher crossing speeds in roundabout, the <i>RDI</i> increases and the FT decreases.	98
4.23	FT (€ per ton.km) vs. <i>RDI</i> values for different runs. For lower absolute value of the deceleration, the <i>RDI</i> decreases on average, but the FT decreases too.	98
5.1	Structure of the driver model generator.	103
5.2	Cycle tuning function principle.	106
5.3	Example of a self-learning cycle on a chassis-dynamometer (vehicle loaded to 14T - slope 0%)	107
5.4	Example of a self-learning cycle in a simulation (vehicle loaded to 14T)	108
5.5	Example of a table of coasting deceleration (vehicle loaded to 14T - slope 0%)	109
5.6	Example of acceleration vs. speed signal acquired during a self-learning cycle (blue), filtered acceleration signal (magenta) and signal fitted by an exponential model (cyan).	110
5.7	View of the user interface of <i>cycle tuning</i>	111
5.8	Simulation results of different roundabout approach behaviors.	114
5.9	Simulation results of different toll approach behaviors.	115
5.10	Example of stop implementation.	117

5.11	Speed signals for an efficient (green) and a non-efficient (blue) driver on the same cycle (red). There are several phases of coasting (7100m, 7800m, 10600m). To ensure readability, only a few km are represented here and not the 125km-long cycle.	118
5.12	Zoom on a roundabout approach: speed signals for an efficient (green) and a non-efficient (blue) driver on the same cycle from an efficient driver (red).	119
B.1	Table of driving events.	128

List of Tables

1.1	Fuel consumption gain announced by truck manufacturers with an eco-driving training	3
2.1	Driving data	20
2.2	Examples of main driving events from literature	21
2.3	Driving event definitions and examples in literature and in our work.	23
2.4	Groups of the driving features from literature and examples. For the complete list of driving features used in related works, see Appendix C Table C.1	28
2.5	Driving events recognition methods	31
2.6	Example of driver parameters to differentiate normal and racing driver in IPG driver software	45
3.1	<i>Throttle</i> = 0 events definition	55
3.2	<i>Throttle</i> >0 events definition	55
3.3	Road type parameters	59
3.4	Driving features for events classification	60
3.5	Classification rate with initial database	63
3.6	Fischer distance between classes	63
3.7	Confusion matrix	65
3.8	Traffic class clustering	65
3.9	Classification rate with final database	65
3.10	Classification results depending on features	67
3.11	Confusion matrix for urban section	70
3.12	Confusion matrix for main road section	70
3.13	Confusion matrix for highway section	71
4.1	Average and standard deviation of <i>RDI</i> for the 4 times with the same driver and the 33 other drivers. It is measured on the first 16km of the cycle, one-way only.	81
4.2	Correlation study between driving features and <i>RDI</i> on extra-urban road	85
4.3	Difference in % between driver types for selected driving features. Reference is medium <i>RDI</i> drivers.	86
4.4	Correlation study between driving features and <i>RDI</i> on highway	87
4.5	Percentage of the driver population having a continuous event, or not, during approaching the toll gate.	88
4.6	Difference in % between driver types for selected driving features on highway road. Reference is medium <i>RDI</i> drivers.	89
4.7	Examples of different parameter combinations used in the simulations	92

4.8	Examples of simulation results. The conditions of each run are presented in Table 4.7	93
4.9	Correlations between the tested driving features and their <i>RDI</i> measured in terms of Pearson's coefficient	94
4.10	Examples of simulation results for three different coasting behaviors	97
4.11	Differences between RDI and FT for an efficient and a non-efficient driver with a reference driver.	99
5.1	Our choices in the three levels of driver models	101
5.2	Simulation results of different roundabout approach behaviors.	114
5.3	Simulation results of different toll approach behaviors.	114
5.4	Stop features in urban areas.	116
5.5	Results of an Urban cycle.	117
5.6	Average speed, f_c , and RDI results for the reference cycle (non-efficient) and the efficient driving cycle.	118
A.1	Truck usages from GTA	125
A.2	Truck usages ACEA	126
C.1	Driving features	130
C.2	Large scale driving patterns recognition methods and results	135
C.3	Small-scale driving events recognition methods and results	136

List of abbreviations

ACEA: European Automobile Manufacturers Association

ADAS: Advanced Driver Assistance System

ADEME: Agence de l'Environnement et de la Maîtrise de l'Energie

AMT: Automated Manual Transmission

AMTPS: Automated Manual Transmission Power Shift

CAN: Control Area Network

CO: Carbon monoxide

EEM: Energy Efficiency Management

EMS: Engine Management System

Fc: Fuel Consumption

FT: Freight Tarrif

GHG: GreenHouse Gas

GMM: Gaussian Mixture Model

GPC: Generator Paralleling Controller

GPS: Global Positioning System

GST: Global Simulation Tool

GSP: Global Simulation Platform

GTA: Global Transport Application

GTWR: Gross Trailer Weight Rating

GUI: Graphical User Interface

GVWR: Gross Vehicle Weight Rating

HC: Hydrocarbons

HD: Heavy Duty vehicle, GVWR > 16 tons

HEV: Hybrid Electric Vehicle

HMP: Heavy Multi Purpose vehicle

IBL: Instance Based Learning algorithm

KDD: Knowledge Discovery in Databases

LD: Light Duty vehicle, GVWR under 7.5 tons

LVD: Logged Vehicle Data

MD: Medium Duty, GVWR from 7.5 to 16 tons

MHD: Medium Heavy Duty, GVWR from 16 to 26 tons

NDS: Naturalistic Driving Studies

NO_x: Oxides of Nitrogen

OEM: Original Equipment Manufacturers

PI: Proportional Integral. See **PID**

PID: Proportional Integral Derivative controller. A PID controller calculates an error value as the difference between a measured process variable and a desired setpoint. The controller attempts to minimize the error in outputs by adjusting the process control inputs.

PM: Particulate Matter

RDI: Rational Driving Index $RDI = \frac{\text{Average speed (km/h)}}{\text{Average Fuel consumption (L/100km)}}$

RMSD: Root Mean Square Deviation between n values observed y_i and model

values m_i : $RMSD = \sqrt{\frac{\sum (y_i - m_i)^2}{n}}$

RST: Digital controller canonical structure. **R**, **S** and **T** are polynomials.

SD: Standard Deviation

SMT: Synchronized Manual Transmission

SVM: Support Vector Machines

VD: Virtual Driver

VLP: Vehicle Lifetime Productivity

Introduction

1.1 Context and motivations of the study

1.1.1 Fuel economy context

The transportation industry represents the highest share in energetic consumption of the European Union, and keeps growing, as can be seen in Fig. 1.1 and 1.2. In this context, the transportation industry has to deal with the need to reduce greenhouse gas emissions and to become less fuel dependent, driven by the awareness of sustainable development and increasing energy costs. As a result, achieving fuel saving and enabling eco-driving becomes a major goal for transportation research. However, the vehicle is not the only factor which has an impact on the fuel consumption. The driver and the choice of an optimized itinerary also play a central role.

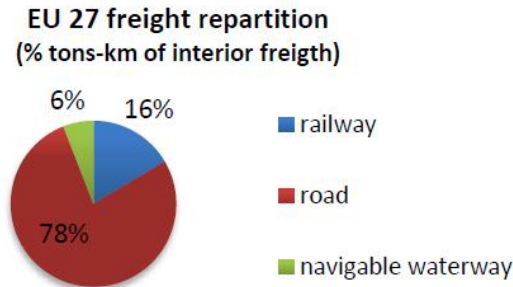


Figure 1.1: Breakdown of freight in the EU. 78% of the freight is transported by the road in 2010.

Even if choosing a more efficient vehicle is important, the impact of driving style is still higher than the impact of a new vehicle¹. This cannot be ignored when one knows that fuel costs can represent 27.2% of the total transportation costs of a company, as presented in Fig. 1.3².

In order to estimate the impact of a driver on the fuel consumption of a vehicle, we performed a data collection with 34 drivers. This study showed a gap ranging from -10.8 and $+18.5$ % between fuel consumption of the drivers on the same cycle with the same heavy vehicle and similar traffic conditions. This result reveals that a

¹Lentreprise.lexpress.fr

²Enquete CNR Longue Distance 2013

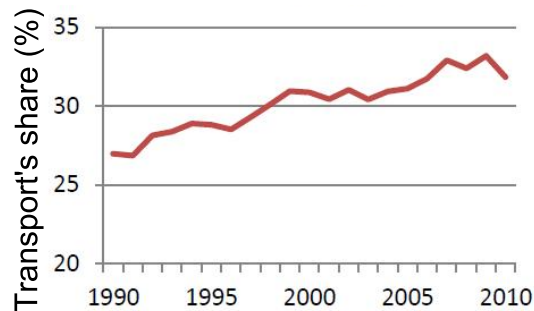


Figure 1.2: Proportion of transportation (%) in the total energetic consumption of EU

change of driver behavior can have significant impact on fuel savings. The 7 Heavy-Duty truck manufacturers on the European market evaluate the impact of driver style after an eco-driving training on average around 10%, as we can see in Table 1.1.

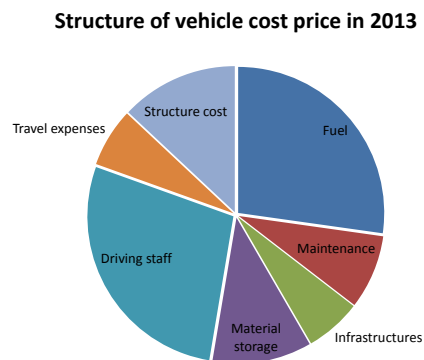


Figure 1.3: Structure of vehicle cost price in 2013. Source: Enquete CNR Longue Distance 2013

1.1.2 Road freight context: a complex challenge

In the European Union, vehicles with a gross vehicle weight rating (GVWR) higher than 3.5 tons are considered as large goods vehicles. We differentiate 4 rolling stocks: rigid, tractor, semi-trailer and trailer.

Vehicles with a GVWR of 7.5 tons are called light-duty vehicles. Medium-duty vehicles range from 7.5 to 16 tons. Heavy-duty vehicles are above 16 tons GVWR.

However the GVWR is not the only criterion differentiating the vehicles. Within the same range of vehicles, trucks can be used for different segments. This is the complexity of the road freight environment. These segments of use are defined within the VOLVO group with GTA, a global language. They are presented in Annex A and are long distance, regional, local and Stop & Go, respectively.

Table 1.1: Fuel consumption gain announced by truck manufacturers with an eco-driving training

Manufacturer	Fc gain
Renault trucks	Up to -15%
Volvo Trucks	-7% to -15%
Daf	-5% to -10%
Mercedes	-8% to -20%
Iveco	Up to -12%
MAN	Up to -10%
Scania	Average -10%



Figure 1.4: Rigid on the left, Tractor on the middle, Rigid and trailer on the right.

Long Distance is described by a mean distance between delivery or pick-up of goods/passengers longer than 50km, few stops, high average speed, and long distances covered during each working period.

Regional usage is described by a mean distance between delivery or pickup of goods/passengers shorter than 50 km but longer than 5km, and a mixture of short and long distances between stops, which are usually clustered.

Local is described by a mean distance between delivery or pickup of goods/passengers shorter than 5 km but longer than 0.5 km. There are many stops, and low average speed.

Stop & Go usage is described by a mean distance between delivery or pickup of goods/passengers shorter than 0.5 km and is characterized by stop-and-go maneuvers, low speed, and stationary vehicle.

The ACEA³ workgroup has a different segmentation: urban delivery, municipal utility, regional delivery, long-haul and construction. Currently, GTA language considers that construction is included in the Stop & Go usage.

For ACEA, long haul usage involves delivery to national and international sites (mainly highway operation and a small share of regional roads).

Regional delivery is the delivery of consumer goods from a central store to selling points (innercity and partly suburban roads).

Urban delivery concerns urban truck operations, such as refuse collection (many

³European Automobile Manufacturers Association

stops, low vehicle speed operations, driving to and back to central base point), whereas municipal utility implies urban delivery of consumer goods from a central warehouse to local stores (innercity, suburban, regional and also mountain roads).

Finally, construction usage is about construction site vehicles with delivery from a central store to very few local customers (innercity, suburban and regional roads; only small share of off-road driving).

These different uses of vehicles must be considered for driver models because they imply different use of actuators (*e.g.*, a driver of a refuse truck will brake and accelerate much more frequently than a long haul driver). The utilization of the truck also involves varying degrees of fuel savings since some phases are more dynamic than others. It will thus be more difficult for a driver to save fuel during a long-haul trip than during a municipal utility usage. However this statement must be tempered, because with undulating roads, fuel consumption can be improved even on a long-haul journey.

1.1.3 A specificity of the truck context: rational driving vs. eco-driving

In addition to the wide scope of study due to the numerous usages of trucks, the road freight context has another specificity, which is the importance of keeping to delivery schedules. Consequently, while we speak about eco-driving for cars, this term cannot be applied in the same way to trucks. Indeed eco-driving means driving in a smarter, safer, more environmentally friendly, more economical and stress-free way; most of these improvements result from more fuel-efficient driving which can often imply a lower speed. Whereas for cars there is an increasingly stringent enforcement of speed limits, and so average speed tends to decrease, trucks are subject to mandatory certified on-board speed-limitors.

Moreover, truck drivers cannot be late because of their delivery deadlines, and the fact that they have to reach their next delivery point quickly because their working hours are regulated. Consequently, for trucks we refer to *rational driving* defined by Maincent [3] as: to exploit a full vehicle's potential to drive more fuel-efficiently while keeping to delivery deadlines. Driving rationally requires the knowledge of the vehicle (especially its dynamics, and also optimal torque yield ranges), anticipating the driving situations, making compromises between average speed and fuel consumption, and using brakes, retarders, and throttle evenly.

Here are some of the rules of eco-driving (for cars) from ADEME [4]:

- Start engine smoothly, and anticipate traffic flow.
- Reduce your vehicle speed. For example decreasing the vehicle speed by 10km/h on highway can improve the fuel consumption by 1L/100km.
- Maintain low engine speeds and shift up early.
- Use assistant devices: ADAS, but also the speed limitation-device, GPS, and traffic information.

- Switch-off the engine at longer stops (or use the automatic start/stop) .

Rational driving is not substantially different and is based on the same general ideas, but the difference is the impact of speed. Maincent [3] explains that rational driving is based on several hypotheses:

- Engine maps have high efficiency operating area.
- Anticipation is a fundamental aspect of driving a vehicle.
- Rational driving is a compromise between average speed and average fuel consumption.
- Fornengo [5] stated that a driver personalizes his/her driving style by his/her control of the brake and throttle pedals.
- According to drivers, driving is based on vigilance, anticipation and attention.
- When compared to usual driving, rational driving seems more dependent on using the brakes and gearbox less.

Rational driving behavior can also be translated in terms of driving strategy by the anticipation of driving situations, an efficient use of the kinetic energy of the vehicle, the right use of engine speeds, and by the crossing speeds of several kinds of road infrastructure, such as roundabouts and curves. Due to the generalization of the use of robotized gearboxes, drivers personalize their behavior especially through the use of brake and throttle pedals. Therefore, the use of correct engine speeds will not be a part of our work. However, the other three factors (anticipation, kinetic energy, and crossing speeds) depend on the driver only, and not on the characteristics of the vehicle, and are therefore interesting points to study.

1.2 Benefits of modeling during the development of vehicles

This work takes place within the VOLVO group, one of the leading worldwide truck manufacturers. The research and development division constantly develops new products, and the creation of representative driver models is an important improvement in the methods. More precisely, I worked within the team responsible for the features verification and validation for vehicle productivity. The team work includes the validation of the fuel consumption and the driveability of future vehicles. The test facilities were provided by the company, as well as the expertise in truck driving. Within this industrial context, the laboratory LIRIS⁴ provides knowledge in data analysis and computer science necessary to discover patterns in the databases.

We showed in the previous paragraphs the importance of the fuel consumption these days. In the road freight context, the average speed also is a key parameter

⁴Laboratoire d'InfoRmatique en Image et Systemes d'information

of the vehicle. Thus, when truck manufacturers test the future vehicles, they need to estimate precisely the fuel consumption and average speed of heavy vehicles. We then try to obtain realistic results in terms of fuel consumption and average speed when we develop the vehicles. These *realistic results* can be obtained by modeling accurately the key components, *i.e.*, the road, the vehicle and the driver, at the early stages of the development process of future heavy vehicles.

Using simulation and modeling during the development of future vehicles has multiple benefits. First, virtual tests have lower costs than real-world tests because a real vehicle with human drivers on the road is more expensive than a simulation. The costs also decrease because the models allow testing at the early phases of a project. Early testing and validating of new concepts is a way of detecting issues at an early stage and thereby gaining time for solving them.

The second gain is the repeatability of the tests. During truck development, repeatability is needed as the same results should be obtained by testing the same vehicle in the same situations. The repeatability is expressed in terms of gaps of the following results: fuel consumption, average speed, and stop distance. The gaps are measured between the same test repeated several times and must be as close to zero as possible.

Simulation and modeling are increasingly used for safety issues and traffic. The development of ADAS also generates new needs in virtual testing. For instance, it is safer to test a lane-keeping ADAS in a virtual world than in a real world in the early stages. Now, in addition to traffic and safety, ADAS for fuel savings can also be tested and developed thanks to simulation.

The development of a realistic driver model for simulation and chassis dynamometer is consequently a relevant research focus to meet global vehicle performance estimation requirements which vary depending on driver behavior. In addition to this, new systems (*e.g.*, ADAS), and new vehicles are more and more complex with many new functions, such as acceleration limiter. These new functions also require improvements in the capacities and relevancy of existing driver models. For example, the fuel saved with an ADAS depends on driver behaviors. Modeling and simulating different driver styles allows us to estimate more accurately the benefits of such ADAS.

1.2.1 Test facilities for modeling

Fig. 1.5 illustrates the different test facilities employed during the whole development process. Development is carried out from a virtual environment to a real one, adding one or more real components at each step, such as engine, gearbox, vehicle, driver, and road.

We have seen that it is important to model realistic driving behavior in order to meet the challenges of current issues in fuel consumption and the truck development process. We aim to implement our model in the two test facilities used during the development process: on our chassis-dynamometer, and on the internal simulation tool.

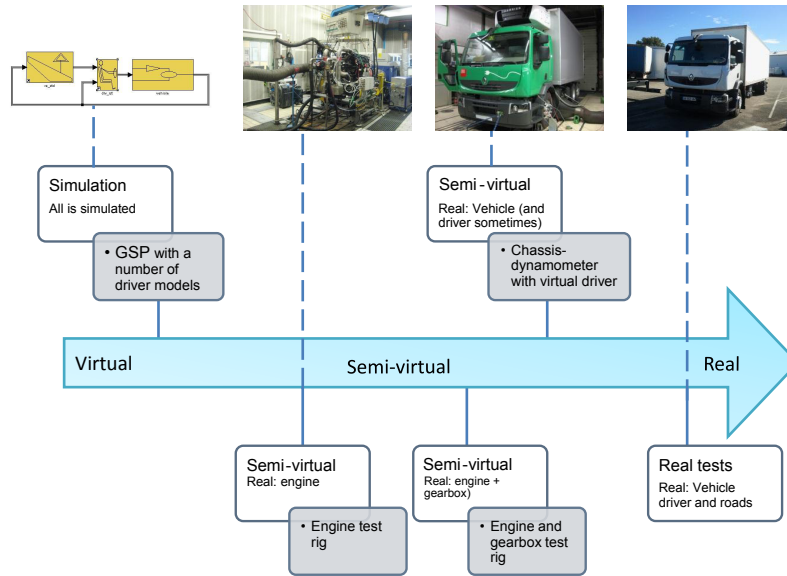


Figure 1.5: All the test means during the truck development process. GSP is our simulation tool presented in the following section.

Concerning simulation, GSP is the Global Simulation Platform used in the VOLVO group. This tool consists of a GUI (graphical user interface) for running the simulation and compiled vehicle concepts. GSP uses Matlab/Simulink to model and simulate different concepts as can be seen in Fig. 1.6. A concept is usually a vehicle driving on a road. However, it could also be an engine rig. The simulation is performed in order to evaluate vehicle fuel consumption and driving performance *etc.* A concept is divided into sub-models for the road and environment, the driver, and the vehicle.

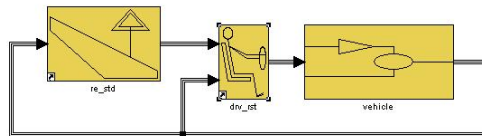


Figure 1.6: Screenshot of GSP simulation tool made with simulink models. The three main components can be seen: road, driver and vehicle.

In contrast to simulation, a chassis-dynamometer is a semi-real environment to test cars or trucks. The real vehicle is placed on rollers which turn when the vehicle is rolling and one can measure the following outputs: power delivered to the surface of the drive roller by the drive wheels, forces, speed of rollers, and so forth. The chassis-dynamometer is a closed loop system since the speed of rollers is also an input to the simulated road. Fig. 1.7 presents the chassis-dynamometer located

on the St-Priest site. The vehicle is connected to a road model, a driver model, and a data acquisition system. Either the virtual driver or a real driver control the actuators, depending on the tests. Fig. 1.8 presents the main components of the chassis-dynamometer and the data exchanges between them. The vehicle is connected to a road model, a driver model, and a data acquisition system.



Figure 1.7: Testing a truck on the chassis-dynamometer

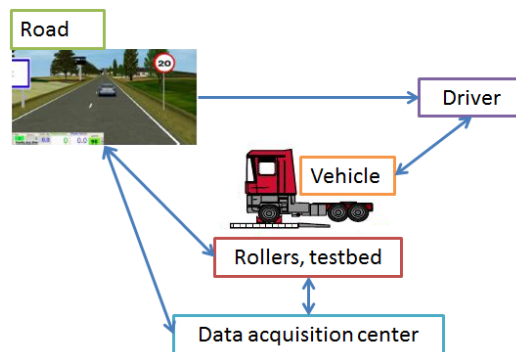


Figure 1.8: Main components of a test on chassis-dynamometer. Arrows represent data exchanges between them.

As previously explained, testing vehicles requires three components, namely vehicle, road, and driver.

In real tests on an open road or on proving grounds, the vehicle, the driver, and the road are real.

In semi-virtual tests, the vehicle (or parts of the vehicle) is real, the road is simulated, and the driver can be real or simulated.

In simulation tests, the vehicle, the road, and the driver are virtual and are based on models.

Fig. 1.9 presents these three components and the exchanges between them. The driver adapts his driving to the road, (*e.g.*, the speed limit), and to the vehicle response, (*e.g.*, the acceleration of the vehicle). The vehicle translates the driver

signals, (*e.g.*, brake and throttle pedals positions), into vehicle response, (*e.g.*, braking or accelerating). Thus the position of the vehicle changes and the new position is sent to the road and the driver with the other outputs of the vehicle. For traffic simulation (*e.g.*, traffic jam, connected infrastructure, *etc.*) the interactions between the components can be different and the development is ongoing.

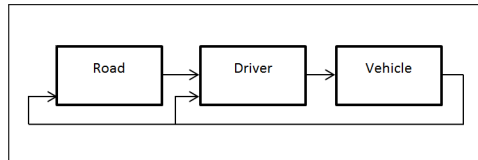


Figure 1.9: The main simulation parts in the simulation tool. Arrows represent exchanges of data between driver, vehicle and road.

We will not elaborate on the vehicle models, but the following paragraphs explain the definition of our road files and the existing driver models. These driver models are actually controller systems which follow speed setpoints.

1. *Digital roads*: a digital road, also referred to as cycle, is defined by four signals:

- Slope profile vs. distance,
- Speed limit vs. distance,
- Stop and stop time vs. distance,
- Road type vs. distance. The road type can be a highway, a regional road, or an urban road.

Some cycles can represent real roads: they are created by digitizing real data, which means that they have the same slope profile, the same speed limits, and the same stops as real cycles. Other cycles can be more artificial. That means that the speed limits, the distribution of slopes or stops do not really correspond to any existing real road.

2. *Driver*: The virtual driver that we use in this work is based on PID controllers. A PID controller calculates an error value as the difference between a measured process variable and a desired setpoint. The controller attempts to minimize the error in outputs by adjusting the process control inputs. This driver model does not need a large amount of data to work: vehicle configuration (Cruise control availability, type of retarders, coupling brake/retarder available), weight of vehicle, speed limit, and current speed, are the only inputs in addition to the road description (slope, distance), and behavior set points (brake deceleration, stop deceleration). Further parameters can be adjusted: deceleration values, acceleration values, brake pedal map, and manual transmission parameters.

The results obtained with this model are repeatable. For instance, stop distances are respected with a gap of $+/- 5m$. For fuel consumption, standard deviation is 0.24% for a test repeated four times. However, it was not designed according to driver behavior, *e.g.*, each braking phase is the same whatever the reason for braking, and there is no anticipation of the braking phases. Similarly, the model is not able for the moment to describe different realistic driving behaviors.

This existing driver model should therefore be improved to enable truck testing in terms of fuel consumption and average speed with respect to various driver behaviors.

1.3 Objectives and issues

Given the fact that the existing simulation tools do not have realistic driving behaviors, the first goal of this thesis work is to create different driver behavior models and implement them on the existing simulation tools, *e.g.*, GSP and chassis-dynamometer, for testing trucks in closer real conditions. Furthermore, given the specific truck context where one needs to achieve fuel saving while keeping to delivery schedule, a second goal of this thesis work is to characterize rational driving in terms of driving features through simulations.

The driver behavior is defined by the use of actuators, *i.e.*, throttle, brake, retarders, to control a truck facing a driving event, *e.g.*, stops, take-offs, roundabouts, speed limits, tolls *etc.* For example, facing a same stop event, a driver can depict an aggressive behavior by hard braking at the last moment[6], or sketch a calm behavior through smooth braking [7], or show a rational behavior in anticipating the stop event, making use of coasting and thereby achieving fuel saving. Implementing driver behaviors on a simulation tool thus implies first to automatically recognize driving events and then to characterize various driver behaviors facing a same driving event. However, in a context of simulation, available data are those in a digital road, thus limited to slope profile, speed limit, stop and stop time, and road type. Furthermore, a driver can approach different driving events in a similar way in terms of in-vehicle data. For example, a roundabout and a give-way can be considered as similar when analyzed in terms of in-vehicle data, *e.g.*, the use of actuators and desired vehicle speeds. In such a context, automatic recognition of different driving events, *e.g.*, stop, roundabout, take-off, represents a major challenge.

Once a driving event recognized, one needs to characterize various driving behaviors facing the same driving event. As drivers behave differently through the use of the actuators to control their vehicle, the challenge here is to find out appropriate driving features based on the use of the actuators to enable to differentiate between driver behaviors. In the literature, driver behavior is studied for different purposes, *e.g.*, safety [6], mental workload [3], fuel consumption [8], *etc.* In this work, we are interested in characterizing driving behaviors for rational driving in the truck context. The first difficulty here is to define a measure of rational driving which en-

ables to compare driver behaviors facing a same driving event, *i.e.*, giving the ability to state that a particular driving behavior is better than another one in terms of rational driving. The second difficulty is then to characterize driving features which lead to a rational driving behavior with respect to various driving events, despite the fact drivers can depict inconsistent behaviors facing a same driving event.

Finally, since our goal is to simulate different driving behaviors in our simulation tool GSP and on chassis-dynamometer, we have to implement the driver models on these two test facilities. The challenge here is to incorporate driver behavior models into the existing testing tools because they have some built-in limitations. For example, the structure of the file describing the road must follow some rules, which impose the format of the setpoints for the controller.

1.4 Approach and contributions

1.4.1 Approach

Automatic recognition of driving events as well as characterization of driver behaviors might rely on heuristic rules. However, given the number of driving features and their possible combinations, the proposed approach to tackle both the problems, as illustrated in Fig. 1.11, is data-mining based and consists of elaborating models through machine learning, *e.g.*, C4.5 or k-NN. Specifically, the following process was carried out to create and implement different driver behavior models onto the in-house simulator GSP and the chassis-dynamometer:

- Automatic recognition of driving events with real data and simulated data.
- Definition of a rational driving index which enables the ranking of driver behaviors depending on the trade-off between the average speed and the average fuel consumption.
- Identification of driving features that enable to differentiate between driver behaviors.
- Implementation of a tool that models different driver behaviors regarding rational driving. The implemented tool makes use of the above results, *i.e.*, learnt models enabling automatic recognition of driving events and those differentiating between driver behaviors.
- Testing and evaluation of the implemented tool onto the in-house simulation tool GSP and the chassis-dynamometer.

Fig. 1.10 gives the flowchart of such a process. The first stage (on the left) aims to learn from driving data a model which enables automatic recognition of driving events, as well as to identify driving features characterizing driver behaviors in terms of rational driving. Once learnt driving event classification model and identified driving features, they are applied to the in-house simulator and chassis-dynamometer

(on the right of the figure) for an accurate estimation of fuel consumption and speed during vehicle tests in simulation and on a chassis-dynamometer.

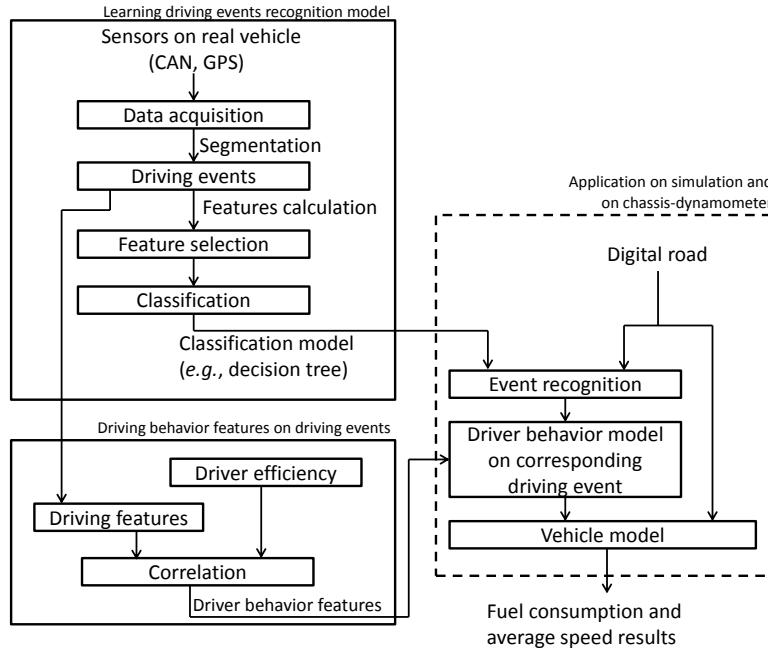


Figure 1.10: Overall approach of our work.

Learning models through data mining generally requires large quantities of labeled data. Some studies analyze driving through naturalistic driving studies (NDS) [9]. NDS offers the advantage of having a large amount of data in recording in-vehicle data when people drive their car in their real life. Unfortunately, when we started this study, we did not have access to such data. Instead, we had access to a large database of aggregated data on average speed and fuel consumption from customers of Volvo, thus missing a lot of useful information, *e.g.*, time coded speed, the use of the actuators to drive the underlying truck, *etc.* This was another challenge that we had to face when we started this thesis work and led us to launch several campaigns of driving data acquisition.

1.4.2 Contributions

The main contributions of this thesis work can be summarized as follows. They will be further developed in Chapters 3, 4 and 5, respectively.

- Automatic recognition of driving events through a set of selected driving features. The state-of-the-art depicted various driving events, *e.g.*, lane changes [9], hard turns [6], traffic jam [10], acceleration [11], *etc.* according to the target application being studied, *e.g.*, safety, traffic analysis, *etc.* In this work, our overall goal is to understand how drivers behave to various road driving events. Therefore, we characterize these driving events according to

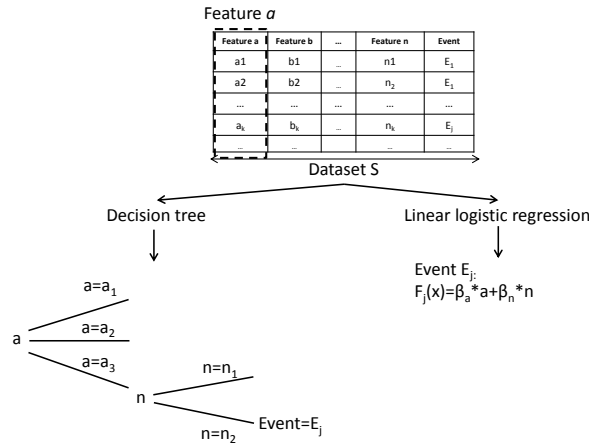


Figure 1.11: Model learning using data mining. We can see the output format of two machine learning algorithms used in this study: on the left the decision tree, and the linear logistic regression on the right. As it can be seen, we can easily interpret these models as they keep the physical sense of the features.

the utilization of the actuators, namely *throttle pedal*, *brake pedal*, *retarders*, and *coasting*. The recognition using two machine-learning algorithms performs recognition of driving events from real data and also from simulated data. This was a key requirement since our results must be integrated in our simulation tools.

- Introduction of a relevant measurement, namely rational driving index (RDI), to rank drivers according to their driving efficiency. In the literature, drivers can be ranked depending on their fuel consumption. However, we go one step further by proposing an indicator which includes the average speed and which is significant to rational driving.
- Identification of driving features correlated with the above rational driving index (RDI), and implementation of these driving features to create different driver behaviors facing a selected driving event.
- Development of a new tool integrating the learnt model for automatic recognition of driving events and identified driving features correlated to rational driving. This tool allows us to test different driving behaviors in our tests facilities. It is a significant work since existing driver models are not always based on real behaviors.

1.5 Thesis Organization

The **second chapter** presents an overview of related works concerning the automatic recognition of driving events and driver styles. In particular, we study the driving features extracted for the machine learning methods and applied to the

recognition of driving events or drivers. Then, this chapter discusses state-of-the-art driver models, whether the models concern fuel consumption or not. This is a broad approach because even if the context is not similar, the methods can be interesting, or adapted to our work.

The third chapter develops classification work for driving events. The goal is to recognize driving situations before starting to identify how drivers may react differently when facing these situations. We first present the data collections which were carried out to build our database. There are many parameters (*e.g.*, weather, traffic), influencing fuel consumption and speed. Therefore, the conditions of the data collection are important to obtain comparable data. Then we present our results obtained with two machine-learning algorithms, namely decision tree and linear logistic regression.

The fourth chapter focuses on the differences in driver behavior leading to differences in fuel consumption and average speed. This work was done in two environments: one extra-urban road with roundabouts, and one highway road with a toll gate. From these two kind of driving data, we correlate some driving features with a Rational Driving Index (RDI), *i.e.*, an index that sorts the drivers in terms of rational driving. Then we simulate some differences in driving behavior in order to quantify precisely the impact of the selected driving features.

The fifth chapter presents the core of the modeling process which was set up for our need of create different driver models in terms of rational driving. We then describe the implementation of different driving behaviors on three driving events, namely roundabouts, tolls, and stops.

Finally, **the sixth chapter** presents the conclusions and prospects that can be drawn from this work.

1.6 List of Publications

Published International Peer reviewed Conference Papers

D'Agostino C., Saidi A., Scouarnec G., Chen L., *Learning-based driving events classification*, Proc. IEEE Int. Conference on Intelligent Transportation Systems (ITSC 2013), 2013

D'Agostino C., Saidi A., Scouarnec G., Chen L., *Rational Truck Driving and Its Correlated Driving Features in Extra-Urban Areas*, Proc. IEEE Intelligent Vehicles Symposium (IV'2014), 2014.

Submitted International Peer reviewed Journal Manuscripts

D'Agostino C., Saidi A., Scouarnec G., Chen L., *Learning-based driving events recognition and its application to digital roads*, Submitted to IEEE Transactions on Intelligent Transportation Systems, Submission Date: Jan., 2014, Manuscript Status: 2nd revision.

D'Agostino C., Saidi A., Scouarnec G., Chen L., *Characterization of rational truck driving through driving features*, Submitted to IEEE Transactions on Intel-

ligent Transportation Systems, Submission Date: July, 2014, Manuscript Status: Submitted.

Knowledge extraction from driving data and driver models

Drivers typically depict different behaviors in different driving situations impacting fuel consumption and average speed. Our data collection carried out with 34 different drivers showed a gap ranging from -10.8 to $+18.5$ % between drivers. Fuel consumption naturally depends on road and traffic conditions. However, these gaps between drivers suggest that driver behaviors faced with various driving events, *e.g.*, approaching a roundabout or a toll gate, have a major impact on fuel savings. In this work, we aim to automatically recognize various driving events, only using in-vehicle driving data, *e.g.*, vehicle speed, in order to further implement different driving styles or behaviors on simulated driving data, *e.g.*, digital roads. In order to achieve the recognition of driving events, we use a database of driving data, and we aim to extract knowledge from these data, *i.e.*, the driving event differences in terms of driving approach, and the different driving behaviors in terms of rational driving on the selected driving events. Once the information extracted from the database, we aim to implement these differences between drivers in our testing tools.

In this chapter, we thus present the principles of knowledge extraction from databases. Then we focus on driving events recognition studies and driving behavior recognition works, and show that our work is quite different from those related studies. On the third section, we present an overview of driver models from related work to review the existing implementation methods for driving behaviors.

2.1 Knowledge Discovery in Databases (KDD) process

As we aim to extract knowledge from driving data, this work is in the field of Knowledge Discovery in Databases. We will briefly present the principles and processes of discovering useful knowledge from data.

The goal of KDD is to discover knowledge by using statistical methods. KDD has been given different names, including data mining, knowledge extraction, data pattern processing, information discovery, data analysis *etc.* For some authors, the term *data mining* is only a step of the process of data analysis, whereas for others [12] *data mining* is the whole process. Whatever the definition, the data analysis process is guided by its final application. Method choice depends on the analysis' purpose or on the type of data available. Before the analysis step, the inputs and

expected outputs are known (*e.g.*, the desired form of the outputs can be rules, probabilities, *etc.*).

The term KDD was coined at the first KDD workshop in 1989 [13] to emphasize that knowledge is the end product of a data-driven discovery [14]. Therefore, the extraction of information, or knowledge, from raw data is a whole process which does not only include the data mining step, *i.e.*, fitting models to, or determining patterns from, observed data. This process was presented by Brachman and Anand [15], emphasizing the interactive nature of the process.

Fayyad *et al.* propose a representation of the basic steps of the process in [14] without detailing the many interactions between the steps. This is only a schematic representation, because these interactions do exist, as the authors clarify them in their work. This representation of the KDD process is presented in Fig. 2.1. This figure does not include the initial phase that states the definition of the objectives for an analysis. The problem and the objectives must be stated clearly to set up the analysis properly. This is one of the most important steps of the process as it determines the methods to be employed.

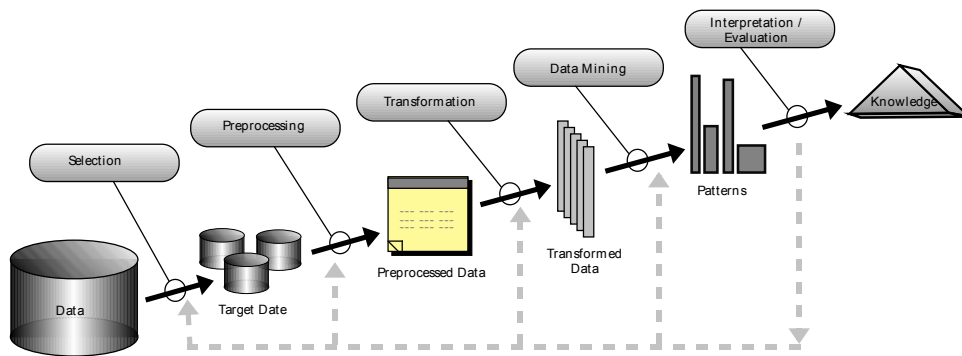


Figure 2.1: An Overview of the Steps That Compose the KDD Process

We will only present the stages represented in this figure. The first stage is the creation of a target data set from raw data: selecting a data set, or focusing on a subset of variables of data samples, on which discovery is to be performed. In general, this data is the fundamental input of the subsequent analysis. The data is usually represented in a table format that is based on the previously established purposes. The second step is cleaning and preprocessing. Basic operations include removing noise if appropriate, collecting the necessary information to model or account for noise, deciding on strategies for handling missing data fields, and accounting for time-sequence information and known changes.

The third stage is data reduction and projection: finding useful features to represent the data depending on the goal of the task. With dimensionality reduction or transformation methods, the effective number of variables under consideration can be reduced, or invariant representations for the data can be found.

Some stages are not represented on this figure: matching the goals of the KDD process to a particular data-mining method, exploratory analysis and model, and

hypothesis selection: choosing the data-mining algorithm(s) and selecting method(s) to be used for searching for data patterns. This process includes deciding which models and parameters might be appropriate, and then matching a particular data-mining method with the overall criteria of the KDD process (for example, the end user might be more interested in understanding the model than in its predictive capabilities).

The fourth step is finally the data mining step, *i.e.*, searching for patterns of interest in a particular representational form, or a set of such representations, including classification rules or trees, regression, and clustering. The user can significantly aid the data-mining method by correctly performing the preceding steps.

Then the fifth stage is interpreting mined patterns, possibly returning to any of the previous steps for further iteration. This phase can also involve visualization of the extracted patterns and models. The evaluation of the performance of the used methods considers diagnostic measures of a statistical type, as well as constraints on the business (*e.g.*, time and resources), and the quality of the data. The following and final step, not shown in Fig 2.1, is acting on the discovered knowledge: using the knowledge directly, incorporating the knowledge into another system for further action, or simply documenting and reporting it to interested parties. This process also includes checking for and resolving potential conflicts with previously believed (or extracted) knowledge.

The KDD process can involve a significant number of iterations and can contain loops between steps. Whereas the data mining step is the core of the process, the other steps are as important (and probably more so) for the successful application of KDD in practice.

We will present in the next chapters our choices and analysis for all these different phases of the data analysis process, and we will compare them to state-of-the art studies. However, it is really important to underline that we chose our input data, and selected algorithms according to these data and to the output format of the results that we aimed to obtain. This is all designed in order to fit to our specific work. Related studies are, such as ours, dependent on the final objective as we will see below.

2.2 Knowledge discovery in driving data

2.2.1 Driving data

There are many studies involving driving data analysis. Therefore, the data, and even the kind of data studied can be really different. Engstrom [16] separates the driving data in three kind of patterns, depending on the length of the driving data: large-scale, small-scale, and real-time scale driving patterns. Large-scale driving patterns, *e.g.*, 5-min speed data, are long-term driving data of a few minutes rather used to analyse global driving events or behaviors. These patterns are often used to determine driving context, *e.g.*, road type [16], identify drivers [17], or estimate fuel

consumption [18]. Miyajima *et al.* [17] use gas and brake pedal patterns modeled via GMM for 5 minutes for the purpose of driver identification.

Small driving patterns consist of driving data typically of a few seconds, which allow extraction of meaningful features during a driving maneuver [9] [6] [11] [7] [19], *e.g.*, changing lane, braking for a roundabout. Finally, the last length is real-time data, *e.g.*, eyes or head position used to classify in-vehicle operations. The order of magnitude of the duration of this kind of data is around a few milliseconds. Table 2.1 summarizes the three kinds of driving patterns.

Table 2.1: Driving data

Related work	Our work
Real-time scale, Small-scale, Large-scale.	Small-scale

Many types of data can also be acquired, coming from different sensors and signal processing systems. The most common sources of data are:

- Vehicle data, *e.g.*, signals from CAN bus [20], from EMS, *etc.*
- GPS data, *e.g.*, latitude, longitude.
- Video data, *e.g.*, filming the road from a fixed point [10], filming the road from the vehicle, the inside of the vehicle, or the driver.
- Map database, *e.g.*, NAVTEQ or Google maps databases provide information linked to the geoposition. The information can be the road type, the altitude, the longitude, the latitude, and so forth.
- Other sensors, *e.g.*, atmospheric pressure sensor, accelerometers, gyroscopes [9] [6] [11] [19].

For the classification of driving events, *i.e.*, small-scale driving data, we will use vehicle data coming mainly from CAN bus and GPS data. Video recording will help for the labeling of driving events. We will specify the conditions of the data collection in Section 3.2.

Therefore, given the goal of our study, we will use small-scale data to define and recognize driving events presented in the following paragraphs.

2.2.2 Driving events definition

As stated before, there are many ways to define driving events according to the objective of the underlying study. As our objective is to understand driver behavior, we are concerned with driving events with respect to which drivers can display

different behaviors, through the utilization of different actuators. We will detail the events definition on Section 3.1.

However, it is important understand at this early stage how we delimit the driving events. To control the speed of his vehicle when facing an event such as roundabout or take-off, a driver uses three different actuators, namely *throttle pedal*, *brake pedal* and *retarders*. Throttle and brake pedals are the pedals that are commonly known, used to accelerate and to brake the vehicle, respectively. Retarders are devices assisting the functions of primary friction-based braking systems. *Coasting* occurs when the driver uses none of the three above-mentioned actuators.

In this study, driving events are delimited according to the utilization of these actuators, and result in two different categories. The first category includes *throttle>0* events, which start each time the driver begins to use the throttle pedal. On the contrary, we observe a *throttle=0* event in the second category when the driver uses the brake and/or retarders in between two adjacent coasting periods.

What can be observed in driving event recognition studies is that the definition of driving events is often associated with the objectives of the aforementioned works. For example, events in studies focusing on safety can be right and left turns, and can be delimited by threshold on the yaw angle [9]. Moreover, most state-of-the-art works focus on traffic and safety issues [6] [10] [11] [9], which leads to the definition of application-dependent driving events. Table 2.2 presents the main types of driving events in the literature.

Table 2.2: Examples of main driving events from literature

Events	References
Turns	[9] [6] [11] [19] [21] [22] [23] [24]
Curves	[9] [21]
Lane change	[9] [6] [23]
Straight across	[19] [21]
Excessive speed/acceleration	[6] [11] [22] [23] [24]
Brake	[6] [11] [22] [23] [24]
Stop	[9] [19]
Roundabout	[19] [21]
Freeways entries and exits	[9]
Lane drifting	[22]
Sharp maneuvers	[23]

Johnson *et al.* [6] present a mobile sensor-platform for intelligent recognition of aggressive driving. This platform aims to distinguish typical (non-aggressive) driving styles from aggressive styles with driver safety in mind, and proposes automatic recognition of 8 driving events, *e.g.*, right turns, hard right turns, left turns, hard left turns, *etc.* The platform uses data collected by various sensors in a mobile phone,

e.g., rear-facing camera, accelerometer, gyroscope, and GPS. In order to detect the beginning of events, they use a simple moving average (SMA) of the rotational energy about the x-axis acquired with the gyroscope. If SMA is greater than an upper threshold, then the event begins until SMA is lesser than a lower threshold. Along the same lines, Eren *et al.* [23] and Paefgen *et al.* [24] also use driving data collected from smartphone sensors to perform driver behavior analysis, *e.g.*, safe driving or risky driving in [23] and critical events in [24]. Alternatively, Karaduman *et al.* [20] rely on CAN bus data for aggressive/calm driving detection.

The aim of Litzenberger *et al.* [10] is traffic flow analysis, *e.g.*, traffic jams, using data acquired by two sensor systems mounted on an overhead bridge above the road. They analyze traffic jam occurrence after deceleration events, and the correlation between the speed of the vehicles, the number of vehicles, the number of lane changes and the time, when a traffic jam occurred. The deceleration events are defined with a threshold, *i.e.*, vehicles with a deceleration $< -5m/s^2$.

Van Ly *et al.* [11] identify three different driving events, namely acceleration, braking and turning. They propose distinguishing two drivers using feature vectors of these driving events, to provide feedback and to reduce the number of dangerous car maneuvers. They define the limits of braking events when the brakelight indicator turns from off to on. The acceleration events are defined by the acceleration pedal position greater than a threshold (*i.e.*, $> 1\%$). The turning events are defined by vehicle speed greater than zero, and the steering angle greater than a threshold (*i.e.*, $> 30^\circ$).

Satzoda *et al.* [9] conduct a more general multimodal study into the possibility of using naturalistic driving studies (NDS) for automated drive analysis. They identify a number of driving events, mainly focused on lateral maneuvers, *e.g.*, right/left lane changes, right/left curves, right/left turns, *etc.* They also define the driving events with thresholds, *e.g.*, if the vehicle center goes beyond a given threshold, *i.e.*, $-1.2m$, a left or a right lane change is indicated.

As we can observe, driving event definition varies considerably from one work to another, depending on the underlying application. In our work, we are concerned with rational driving, *i.e.*, driving events which highlight how a truck driver controls his vehicle. As a result, we define driving events that are quite different from state-of-the-art ones. However, despite these differences, the driving features and the methods applied in these studies could be applicable as we will see in the following sections.

Table 2.3 illustrates the difference between the definition of our driving events and those of related studies. In Appendix C Table C.3 proposes a more comprehensive list of the driving events recognized in these works.

2.2.3 Definition of driving styles

Another type of information that can be extracted from driving data is the recognition of driver behavior. In this field, studies focus either on driver identification,

Table 2.3: Driving event definitions and examples in literature and in our work.

Related work	Our work
Definition	
Defined on the direction of the vehicle, on the acceleration pedal threshold, on steering angle threshold, brake light status off to on .	Defined on the driver's control on actuators of the vehicle.
Examples of events	
Left turn, right turn, straight line, lane change, roundabout, stop...	Roundabout, take-off, toll, exit ramp, maintain speed limit, stop...

i.e., recognizing a specific driver [17], or on driver style recognition [6], *e.g.*, recognizing aggressive *vs.* non-aggressive maneuvers. Driver identification and driver style recognition is particularly useful for ADAS. This is due to the fact that the advice that is provided can be adapted depending on the previously learned driving behavior corresponding to a specific driver. Our purpose is to be able to recognize driving behaviors in the previously recognized driving events. Driver recognition can be useful for many purposes, such as ADAS, simulation and so forth. The recognition of driver styles will be useful for the estimation of realistic fuel consumption and average speed during truck development.

Whereas driver identification only focuses on learning a specific driver behavior, without necessarily labeling this behavior, driver style recognition first requires the definition of these *driving styles*.

In the literature, authors distinguish generally between 2 and 5 classes of drivers. We summarize below the different classes encountered in related works. Even if the final objective of some of these works is not the classification of drivers (in terms of data-mining and machine-learning methods), they do propose different classes, not always linked to driving data analysis.

Doshi and Trivedi [25], and Johson and Trivedi [6] propose a separation of aggressive and non-aggressive behaviors. This separation is really relevant for safety issues. The two groups of drivers clearly tend to behave in different ways in similar situations. They find that non-aggressive drivers are quantifiably and significantly more compliant to feedback from ADAS. However, they also find that the population of non-aggressive drivers needs to be further split in order to detect more significant behavioral trends. Canale and Malan [7] also focus on aggressive driving, and distinguish three driving styles: normal, quiet and aggressive. The framework of this work is an adaptive cruise control (ACC) with stop and go features for use in urban areas. In order to adapt the control strategy to the driver style, a number of different drivers are studied through the statistical analysis of their behavior while

driving.

The purpose of Fancher *et al.* [26] is to examine manual driving style to compare manual driving with ACC driving. They present two methods to rate drivers and their driving style. The first method is the percentage of time, expressed as the probability of a driver to spend time in the near region of the range-versus-range-rate space. However, they do not provide a detailed understanding of the driving style of each individual. Based on their initial ideas [27], an expanded classification scheme is developed for driving styles at highway speeds, *i.e.*, above 88km/h . They measure the probability of the driver to be fast, far, close and slow by using the vehicle velocity and the distance between the preceding vehicle.

The quantity A is a measure of the *far* tendency of a driver; B represents the *fast* tendency; C represents *close*; and D represents *slow*. The 75th percentiles for values of A, B, C, D, and the products AB, BC and AD, are determined by examining the data for all drivers. This information is used to classify drivers to provide a descriptive portrayal of five types of driving styles.

“Ultraconservative” means that AD or D is greater than the 75th percentile. Ultraconservative means an unusual tendency towards far and/or slow driving.

“Planner” means that AB or B or A is greater than the 75th percentile. Planner means an unusual tendency towards far and/or fast driving.

“Hunter/tailgater” means that BC or C is greater than the 75th percentile. Hunter/tailgater means an unusual tendency towards fast and/or close driving.

“Extremist” means that the driver satisfies more than one of the above tendencies.

Finally, “flow conformist” means that the driver satisfies none of the above. A flow conformist tends to travel at the same speed as other cars and at approximately the median headway time gap.

These five classes are also used by Abou [28]. However the objective of their work is not to classify drivers, but to propose a probabilistic approach to estimate the variation of accelerations and decelerations in traffic networks as a function of speed and road type. Delorme and Song [29] focus on 2 categories within these five classes, which are the “hunter/tailgater” versus the rest of the drivers.

The explanation is that both Planner and Ultra-Conservative drivers are identified depending on situations in which the preceding vehicle is a long way ahead. These situations are not considered in their study. They also adapt the boundaries for fast/slow probabilities, therefore the transfer of this classification is limited to two groups. Consequently their objective is to compute the different parameters studied, *i.e.*, mean and minimum time to collision, mean time gap, error, critical and comfortable time gap, and break time gap, for each driver type. Finally “hunter/tailgater” and the “ultra-conservatives” are the two extreme groups, as the “Hunters” always display the smallest values whereas the “ultra-conservatives” have the highest ones (for each parameter).

Although this classification is interesting, it is based on the way the preceding

vehicle is followed and is thus not really adapted to our work where we are concerned with fuel consumption. A work which is closer to ours is Maincent's [3] study in which she proposes three classes of driver behaviors: *efficient*, *moderately efficient* and *inefficient*, in terms of rational driving.

In works focusing on simulation, *i.e.*, reproduction of different driver behaviors but not always linked to real driving data analysis, they also define different driver types. Zorrofi *et al.* [30] present a study of the impact of the driver's driving habits on the fuel saving of a hybrid transit bus using computer simulations. In simulation, they distinguish 3 categories of driver: *mild*, *normal* and *aggressive*. The mild driver accelerates and brakes mildly. The normal driver immediately follows the speed limit instructions without excessive application of acceleration and braking. The aggressive driver follows the speed commands by sharp and abrupt acceleration and braking. The aggressive driver's behavior resembles an on/off control system with a narrow tolerance band around the reference. Kiencke and Nielsen [31] propose a whole automotive control system and have differentiated beginner, professional and fast drivers without further explanation.

We have seen that drivers can be classified depending on their speed, fuel consumption, distance and time between the preceding vehicle, and their aggressiveness, which can be expressed by higher acceleration values (lateral or longitudinal). We will propose to classify drivers in three classes, efficient, medium and non-efficient drivers in terms of rational driving, such as in [3].

2.2.4 Driving features

In the literature, driving features are extracted for various purposes and from various data patterns. Different goals include the aforementioned driving events, but also road type recognition [16], driver classification [17], recognition of aggressive driving [6], or fuel consumption estimation [18]. Our first aim here is to recognize driving events with respect to rational driving, *e.g.*, stop, take-off, *etc.* We thus need to extract meaningful driving features relevant to the driving events being studied.

We list all the driving features used with large-scale and small-scale driving patterns in Appendix C Table C.1. These parameters can often be grouped in different themes. We categorize the driving features into eight different groups. These different groups are presented in Table 2.4 with examples of the most common driving features observed in related works.

The first group relates to various vehicle speed statistics, (*e.g.*, average speed, speed distribution), whereas the second group relates to vehicle acceleration, (*e.g.*, average acceleration when acceleration is greater than a threshold).

The third group consists of noticeable sub-parts, which describe a precise part of the driving data and compute different proportions of driving periods, *i.e.*, proportion of standstill time, percentage of time with constant speed, *etc.*

The fourth group is entitled *Independent* and consists of features that do not depend on the vehicle, such as driving duration and number of stops per kilometer.

The fifth group mainly concerns analysis of energetic power.

The sixth group relates to various parameters concerning engine speed and gear shifting, while the following group collects information on lateral driving, *i.e.*, lateral and swing accelerations or jerks.

Finally, the last group concerns driver control of the vehicle with the actuators.

The features used for our work are presented in Chapter 3 Table 3.4.

Driving events recognition works [9] [6] [11] [19] compute statistical features over lateral acceleration and/or curvature data, acquired using accelerometers, gyroscopes or GPS. Steering angle and position of the vehicle on the current lane are also common attributes for safety studies. In our work, we focused on longitudinal dynamic features as one of the main aims of this work are the simulation and the chassis-dynamometer neither of which includes lateral efforts. Therefore we will focus on driving speed, acceleration and environment-related driving features. Consequently, even if we defined driving events that are quite different from state-of-the-art ones, the various features proposed in these state-of-the-art works can be useful for automatic recognition of our driving events.

In driver recognition works, Canale and Malan [7] study the driver's style via the statistical analysis of their behaviors, *i.e.*, the mean value and the standard deviation of a vehicle and environment data, *e.g.*, relative speed with respect to the preceding vehicle, slope. However, their study is limited to stop and go phases on urban sections. Johnson *et al.* [6] propose recognition of aggressive and dangerous driving actions. Actually it is the recognition of the maneuvers (*e.g.*, left turn or aggressive left turn) that determines whether the driver is aggressive or non-aggressive.

Driving features can also be used for correlation, *e.g.*, with fuel consumption, instead of classification. If we focus more specifically on works concerned with fuel consumption, it appears that related research works so far are mostly focused on economical driving and not on rational driving. Lin *et al.* [32] categorize various factors impacting the fuel consumption into four classes: vehicle, road, environment and driver. The specific impact of the driving style or behavior on the fuel consumption can thus be studied by isolating the driver independent parameters and making them stable.

Therefore, given a similar external environment, *i.e.*, road, traffic, weather, we can identify the key driving features that differentiate driving behaviors. Liimatainen [33] follows this method and proposes a fuel consumption measurement based on groups formed by runs with similar vehicles, routes, and times of day. The individual driver's average fuel consumption in a specific group is then compared with the average fuel consumption of all drivers in that specific group.

Ericsson [8] lists a set of driving features to study the effect on fuel utilization and exhaust emission factors. These driving parameters cover all of the aforementioned groups of features. While all these features are interesting for the purpose of fuel utilization in their study, they are not all adapted to our driving events recognition

process. For instance, engine related features, *e.g.*, % of time at engine speed $< 1500rpm$, are specific to car characteristics while our aim is to recognize driving events independently of a truck's physical engine properties. In addition, gears and speed features should also be adapted to trucks as vehicle dynamics are different for cars and trucks, even for different kinds of heavy-vehicles. Brundell-Freij *et al.* [34] identified seven factors which have significant effects on the fuel consumption and exhaust emissions. Those features are related to acceleration, stop, speed oscillation, extreme acceleration, late gear changing from 2nd and 3rd gear, engine speeds $> 3500rpm$ and moderate engine speeds in 2nd and 3rd gear.

However, these features are also related to car driving. They are thus not always adaptable to the context of trucks. This is because cars and trucks do not have the same range of values for the engine speeds and gear changing. Moreover the use of a robotized gearbox is a standard for trucks. For the study of truck driver behaviors, it is thus necessary to isolate the factors specific to the driver, *e.g.*, the use of the throttle, brake, from those which are specific to the gearbox management and the vehicle (engine speeds, gear changing, fuel map...). Nevertheless, most of these driving features can still be used for the case of trucks, but with different thresholds, *e.g.*, percentage of time during which the acceleration is greater than a threshold. Montazeri *et al.* [35] list 19 driving features, including in particular most features already proposed in previous studies, *e.g.*, those in [8]. In their study, they define an *effectiveness index* (EI) to evaluate the impact of each driving feature with respect to the fuel consumption and concluded that the fuel consumption and exhaust emissions depend on energy, mean velocity and displacement more than other features. While the introduction of the EI indicator is original and interesting, this index only considers the fuel consumption and emissions but ignores the average speed of a vehicle.

A work within a context similar to ours, *i.e.*, truck driving on extra-urban roads, is made by Maincent [3] who proposed characterizing rational driving through several features: the anticipation of driving events, the use of the kinetic energy of the vehicle, the use of the most efficient engine speed and finally the crossing speed on some road infrastructures, including in particular curves and roundabouts. However, the aim of Maincent's study is to analyze drivers' mental load and the generalization of her conclusion is based on the analysis of driving data from only four drivers.

2.2.4.1 Driving parameters from ADASs

Although ADASs might appear a little far from the main subject, their driving evaluation characteristics can provide a new point of view and may help find criteria to define drivers driving style.

We analyzed 29 systems for both cars (13) and trucks (16). The following is a list of the various kinds of information that can be provided or tasks that are carried out depending on the system: a visualization of how fuel efficient a driver is

Table 2.4: Groups of the driving features from literature and examples. For the complete list of driving features used in related works, see Appendix C Table C.1

Group	Driving parameters
Speed	Continuous, Average, or S.D. of speed [8] [16] [9] [10] [21] [23] [20] Average driving speed (w/o stops) [8] Distribution of speed [8] Frequency of oscillations of the speed curve per 100s.[8]
Longitudinal Acceleration	Continuous, Average, or S.D. of acceleration [8] [6] [19] [21] [22] [23] [24] Distribution of acceleration and deceleration [8] Average number of acceleration-deceleration changes within driving period. [8]
Noticeable sub-parts	Proportion of standstill time (speed<3km/h $ a < 0.1m/s^2$) [8] Proportion of acceleration time ($a > 0.1m/s^2$) [8] Proportion of time at constant speed ($ a < 0.1m/s^2$) [8] Proportion of deceleration time ($a < -0.1m/s^2$) [8] Percentage of time when acceleration exceeds $2.5m/s^2$ [8]
Independent	Number of stops per kilometer [8] Slope [7] [21] Maps information [9]
Energetic-power analysis	Positive kinetic energy [8] Relative positive acceleration[8] Variables representing surrogates for inertial power and drag power [8]
Engine	Parameters concerning engine speed and gear shifting such as idle period [8] [16] [20]
Lateral	Continuous or statistical values of: lateral and swing accelerations, or jerks, and curvature of the road [6] [11] [9] [19] [21] [22] [23] [24]
Actuators	Steering wheel angle [16] [23] Continuous or statistical values of brake pedal [17] [16] Continuous or statistical values of throttle pedal. [17] [16]

in each part of the trip by providing a detailed map with colors and indications (2 systems); the amount of money they saved by using the specific system (4 systems); real-time messages about how to improve their driving performance by visual or audible advice (5 systems); messages at the end of the trip or in the report analysis (5 systems); driving tutorials in order to improve fuel-efficient driving habits (2 systems); disengagement of the transmission and engine to reduce friction losses (3 systems); vehicle speed limitation (maximum, cruising, downhill - 4 systems);

detection of the crest of the hill without GPS (3 systems); information about the engine characteristics (3 systems); and finally real-time data transmission between the vehicle and fleet managers (2 systems).

Regarding the parameters used to evaluate driving efficiency, the acceleration is the driving factor that most constructors (27 in total, 14 for trucks) analyzed in their systems. The acceleration parameter includes various data from the accelerator pedal position to the way the driver presses it. The speed is the second factor most evaluated: 24 constructors including 14 for trucks (24 for vehicle speed, 14 for engine speed). The speed factor involves even driving as well as the maximum speed reached by the vehicle. The braking-deceleration factor is considered in 23 systems (11 for trucks). The other driving factors analyzed are: gear management (15 in total, 9 for trucks), idling (8 in total, 6 for trucks), coasting and driving behavior in hills (6 in total, 5 for trucks), driver's anticipation capacity (5 in total, 3 for trucks), use of retarders or exhaust brake (4 in total, 4 for trucks) and drag (2 for cars only).

These parameters do not require the addition of new parameters to the study, especially because it is difficult to know the precise driving feature behind the generic name. For example, under the *acceleration* heading, it is not always clear which threshold is analyzed by the ADAS. However, if we remove the vehicle-dependent parameters (idling, engine speeds), this analysis crosschecks with the principles of rational driving (except that the driver rating is calculated by only using the fuel consumption and not the average speed) as the ADASs focus on anticipation, acceleration and smooth driving, coasting, and use of retarders and brakes.

2.2.5 Machine-learning methods and results

Machine-based recognition of driving events or driver behaviors requires appropriate learning algorithms. Johnson *et al.* [6] propose recognition of aggressive and dangerous driving actions. They use k-Nearest Neighbors (kNN) and achieve an accuracy rate ranging from 77% to 91%, with a false positive rate of 5% when using different subsets of features. When using the feature vector containing the longitudinal acceleration, which is the closest feature vector to our work since we focus on longitudinal efforts, they obtain a classification accuracy of 77%. Better results are obtained with gyroscope values, lateral acceleration and Euler angle rotation.

Mitrovic [21] proposes to recognize seven common types of driving events, namely driving a vehicle along left and right curves, turning a vehicle left and right on intersections, with and without roundabout, and driving straight across an intersection with a roundabout, for safety purposes. They applied a learning approach, *e.g.*, discrete HMM, and used in-vehicle driving data through symbolic coding. In comparison, our work proposes recognizing a number of driving events of up to 13 classes. Our learning approach is decision tree-based, and the resultant learnt models can easily be interpreted and implemented on digital roads for realistic assessment of fuel consumption and average speed in simulation.

An interesting study is the recognition of powered two-wheelers riding patterns

by Attal *et al.* [19]. They compare five well known machine-learning techniques, namely the Gaussian mixture models (GMMs), k-NN, Support Vector Machines (SVMs), Random forests (RFs) and both discrete and continuous Hidden Markov Models (HMMs). The results have a spread between 60% and 96% depending on the method and the driver. However, for a given driver, all these methods obtained close results, but HMMs approach attained better scores in terms of average rate and standard deviation. HMMs produce results sometimes lower (up to -4.2%) or sometimes higher (up to +15.8%) than accuracies obtained with RF and k-NN, depending on the driver.

In driver identification works, Van Ly *et al.* [11] use feature vectors with statistic values of lateral acceleration, and a gyroscope signal for each braking, turning and acceleration event. They aim to build a driver profile and compare two learning methods, namely SVM and k-mean clustering, to differentiate two drivers (driver A or B) for each event. They achieve classification accuracies ranging from 0.54 to 0.85 depending on events, feature vectors and the percent of test data used in their cross validation. They are then able to implement a majority vote system where if the events are classified 65% of the time as driver A, they count all driving events in that session as driver A at the end of the drive.

Canale and Malan [7] analyze and classify human driving behavior in an urban environment. They use kNN for recognition of driving styles but do not report classification accuracies.

Miyayima *et al.* [17] used the cepstrum GMM-model and the 5-minutes gas and brake pedal histograms to distinguish drivers. They also extract dynamic features of these cepstrums and histograms. They compare the models learnt using GMM with those learnt using log-likelihood, and they obtain results from 47.5% to 89.6% depending on the features.

The following list reminds briefly the principle of each algorithm used in related works.

- A Hidden Markov Model (*HMM*) [36] is a statistical model particularly used for pattern recognition such as speech recognition. The system being modeled is supposed to be a Markov process with hidden states. As opposed to observable Markov model, states are not directly observable in the Hidden Markov Model, but output is visible and depends on the states. A non-Bayesian hidden Markov model can be described by states, observations, emission probabilities (from state to observation) and transition probabilities (from state to state).
- A Gaussian Mixture Model (*GMM*) [37] is a mixture of Gaussians, *i.e.*, a combination of Gaussian densities that is used to model an input. Given a Gaussian Mixture Model, we use the Expectation-maximization algorithm (EM) to maximize the likelihood function of a parameter vector with respect to the Gaussian densities parameters.

- k-nearest neighbors algorithm (*k-NN*) [38] is a classification and regression algorithm presented in Section 2.2.6.3. k-NN computes a distance between instances in order to estimate the closest *k*-neighbors of an object. Then k-NN yields the class membership of that object by a majority vote of its *k* neighbors; the object being assigned to the most common class among the neighbors.
- *k-means* [39] is a clustering method that aims to partition observations into clusters. Each observation belongs to the cluster with the nearest mean. K-means is a basic example of the EM algorithm application.
- Support vector machines (*SVM*) [40] are supervised learning models of a representation of the inputs as points in space, mapped so that the categories are separated by a margin that is as wide as possible. The input vectors can be non-linearly mapped to a higher-dimension feature space. A linear decision surface is constructed in the feature space. Thus the classifier is a hyperplane that separates observations in two categories in the high-dimensional feature space .
- Random forests (*RF*) [41] are a combination of a multitude of tree predictors (*e.g.*, decision trees as we present in Section 2.2.6.1). Each tree is constructed with a random and iid (independent and identically distributed) vector of the learning set. After a large number of trees is generated, they vote for the most popular class.

All these studies consistently show that it is possible to automatically recognize significant meaningful driving events. As can be seen from this overview, standard state-of-the-art learning algorithms are used in these works. In our work, we have given priority to clarity of classification results, and have chosen decision tree and linear logistic regression algorithms. These algorithms are developed in Section 2.2.6.

Table 2.5 summarizes the different machine-learning methods used for driving events recognition, and the methods that we have chosen.

Table 2.5: Driving events recognition methods

Related work	Our work
k-Means, k-NN, RF, GMM, SVM, DHMM and CHMM.	C4.5, LLR, k-NN for comparison.

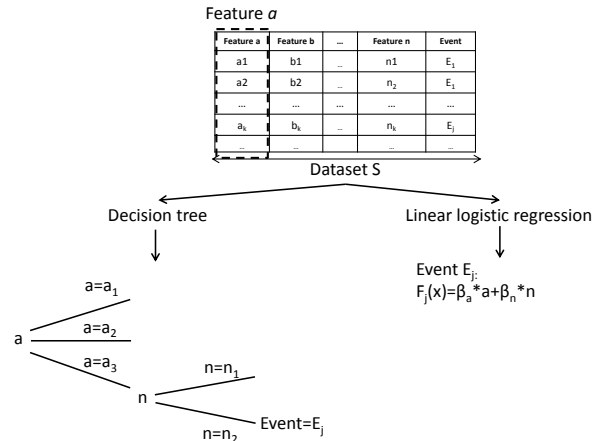


Figure 2.2: Explanations of algorithms utilization. We can see the output form of the two algorithms: on the left the decision tree, and on the right linear logistic regression. These models are observed to be easily interpretable as they retain the physical sense of the driving features.

2.2.6 The selected learning-based algorithms

Once driving events have been defined and discriminating features identified, we now present the machine-learning algorithms for the recognition of driving events. We have seen that in related works, commonly used algorithms are used to recognize driving events. Similarly, we will apply two widespread machine-learning methods, namely *decision tree* and *linear logistic regression*, presented below. We will also present in the following paragraphs the *k-NN* method. This method is often used in state-of-the-art studies, and we will compare our results with those obtained with *k-NN*.

Fig. 2.2 presents the classification process from a raw dataset S to the results of the algorithms. Each instance of the dataset S represents an event Ev and its corresponding values for each feature. As it can be seen on this figure, the output format of the classification can be easily re-used in the next steps of our works (see Fig. 1.10 in Chapter 1).

2.2.6.1 Decision tree

This method is chosen because it retains the physical sense of attributes, extremely necessary for this work. Indeed, one of our priorities is to fully understand why some instances are classified in one class and not in another. Moreover, this method allows us, via the output tree, to clearly see which features are the most important.

The decision tree is one of the most widely used models that partitions the input space into cuboid regions, and then assigns a simple model to each region [42]. The model selection process is a sequential decision-making process, corresponding to the side-view of a tree (*i.e.*, split into several branches at each node). The first step divides the whole input space into two or more regions according to the result of

the test of the model, *e.g.*, the highest gain ratio, the cross entropy, the Gini index. This creates subregions, each of which can be subdivided independently, and so forth. The stopping criterion, *i.e.*, when to stop adding nodes, is commonly based on the number of data points associated with the leaf nodes, and then the resulting tree is pruned back. The twofold objective of pruning is to reduce the complexity of the final classifier as well as improve the predictive accuracy by the reduction of overfitting and removal of sections of a classifier that may be based on noisy or erroneous data. Pruning is based on a pruning criterion that balances residual error against a measure of model complexity.

Among these decision-tree frameworks, we use *C4.5* algorithm in WEKA [43] data mining software to classify the events. The method is called J48 and is based on Quinlan's *C4.5* algorithm [44]. *C4.5* is an extension of Quinlan's earlier *ID3* algorithm [45]. The *C4.5* system consists of four main programs: the decision-tree generator, the production-rule generator, the decision-tree interpreter and the production-rule interpreter.

The decision-tree generator builds the tree structure in which each internal node represents an attribute test, each branch represents a test outcome and each leaf node represents a class label. The *C4.5* attribute test is the calculation of the information entropy, in order to find the attribute that most effectively splits the samples into subsets. The algorithm framework is presented on the following page as Algorithm 1.

The splitting criterion is the normalized-information gain. That means that at each node the most discriminating attribute is determined depending on the gain-ratio, *i.e.*, the information-gain weighted by the number of instances in each class (split information). The attribute with the highest information-normalized gain is chosen to make the decision. The gain $G(S,A)$ for attribute A of the set S before the split on A is:

$$G(S,A) = E(S) - \sum_{i=1}^m f_s(A_i) \times E(S_{A_i}) \quad (2.1)$$

where m is the number of different values of A in S , $f_s(A_i)$ is the frequency of the items possessing A_i as value for A in S , A_i is the i^{th} possible value of A , S_{A_i} is a subset of S containing all items where the value of A is A_i and $E(S)$ is the information entropy of the dataset S before the split on A :

$$E(S) = - \sum_{j=1}^n f_s(j) \times \log_2 f_s(j) \quad (2.2)$$

where n is the number of different values of attributes in S and $f_s(j)$ is the frequency of the value j in the set S .

To decide if a node should become a leaf or not, *C4.5* estimates the error rates using a confidence level. The definition of this parameter is not explicitly given in

Algorithm 1 Tree Growing and Tree Pruning framework algorithms

Tree Growing (S, A, y). Create a new tree T with a single root node.

Require : S - Training Set, A - Input Feature Set, y - Target Feature

if One of the Stopping Criteria is fulfilled **then**
 Mark the root node in T as a leaf with the most common value of y in S as a label.

else
 Find a discrete function $f(A)$ of the input attributes values such that splitting S according to $f(A)$'s outcomes (v_1, \dots, v_n) gains the best splitting metric.
if best splitting metric $>$ threshold **then**
 Label t with $f(A)$
for each outcome v_i of $f(A)$: **do**
 Set $Subtree_i = TreeGrowing(\sigma_{f(A)=v_i}S, A, y)$.
 Connect the root node of t_T to $Subtree_i$ with an edge that is labelled as v_i

else
 Mark the root node in T as a leaf with the most common value of y in S as a label.

return T

Tree Pruning (S, T, y). Pruning the tree T .

Require : S - Training Set, T - The tree to be pruned, y - Target Feature

repeat
 Select a node t in T such that pruning makes maximal improvements to evaluation criteria
if $t \neq \emptyset$ **then**
 $T = pruned(T, t)$

until $t = \emptyset$

return T

[44], but the expression can be found in [46] as shown below.

$$CF = \sum_{x=0}^{Er} \binom{N}{x} p^x (1-p)^{N-x} \quad (2.3)$$

Where CF is the confidence factor, N is the number of training instances, Er is the number of errors, and p is the estimated probability of error. Beck *et al.* [47] discuss how the decision tree produced by $C4.5$ is affected by the change of the $C4.5$ default parameters, such as CF (confidence factor) and MS (number of minimum split-off) cases. They explain that CF can be interpreted as the probability that there will be Er or fewer errors in N unseen cases, after we observe Er errors in N training examples from a node. Therefore, with known CF , N and Er values at a node, the equation is solved for the parameter p . The confidence factor is used for pruning in the $C4.5$ algorithm. Thus, the lower the value of CF , the more pessimistic the pruning will be because the error rate p will increase, *i.e.*, smaller values of CF incur more pruning. For example, if $CF=25\%$, the 75% confidence interval of the error rate on unseen cases is $[0, p]$.

Therefore, when using $C4.5$ algorithm in WEKA, the main parameters are the confidence factor, the minimum number of instances per leaf, and the amount of data used for reduced-error pruning (one fold is used for pruning, the rest for growing the tree).

Fig. 2.3 displays an example of decision tree visualization.

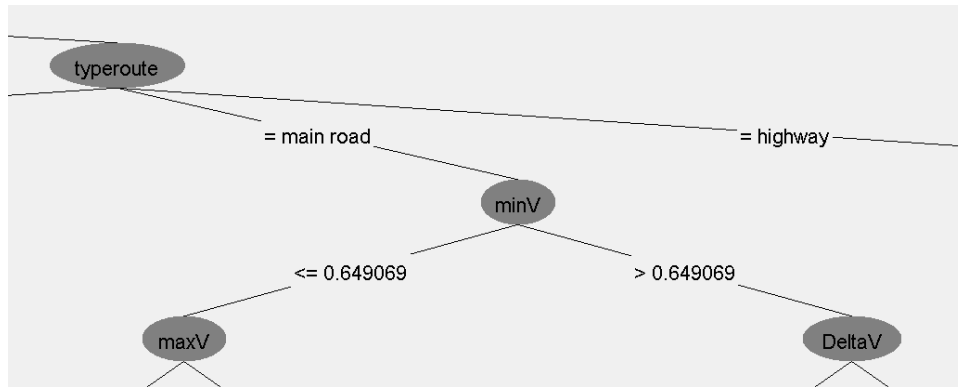


Figure 2.3: Example of visualization of a part of a decision tree.

2.2.6.2 Linear logistic regression based classifier

We also use *linear logistic regression* (LLR) for comparison with decision trees. LLR builds a linear model using the feature values and retains their physical sense, such as $C4.5$ algorithm. Regression variables are the features listed in Table 3.4. Moreover, each feature is weighted in the process of building the model and selecting important features.

In WEKA software, LLR is referred to as the *SimpleLogistic* algorithm, and is based on [48] and [49]. LLR creates linear logistic regression models to obtain the posterior probability of a given instance to belong to the class. The linear logistic regression associates a linear model with a probability function, and ensures that the probabilities sum to one and remain in $[0, 1]$. According to [48] and [49] the linear logistic regression model is defined as follows:

$$Pr(Ev = k | Features = a) = \frac{e^{F_k(a)}}{\sum_{l=1}^K e^{F_l(a)}} \quad (2.4)$$

where

$$\sum_{l=1}^K F_l(a) = 0 \quad (2.5)$$

$$F_k(a) = \beta_k^T \cdot a = \sum_{i=1}^n \beta_{ik} a_i + \beta_{0k} \quad (2.6)$$

where $Pr(Ev = k | Features = a)$ is the posterior probability of belonging to class k given the input feature vector $a = (a_1, \dots, a_n)$, where n is the number of features, K is the number of classes, and $F_j(f_k)$ is the linear model with parameters β_j . The simple logistic algorithm estimates these parameters based on the LogitBoost algorithm [50], which performs forward stage-wise fitting of additive logistic regression models, which generalize the above model to $F_j(x) = \sum_m f_{mj}(x)$ where the f_{mj} can be arbitrary functions of the input variables that are fitted by least squares regression. Numeric optimization algorithms that approach the maximum likelihood solution iteratively are used to find the estimates for β_j . One such iterative method is the LogitBoost algorithm [51] given in Algorithm 2. The variables y_{ij}^* encode the observed class membership probabilities for instance x_i (recall that y_i is the class label of instance x_i), *i.e.*

$$y_{ij} = \begin{cases} 1 & \text{if } y_i = j, \\ 0 & \text{if } y_i \neq j \end{cases} \quad (2.7)$$

The $p_j(x)$ are the estimates of the class probabilities for an instance x given by the model fit so far.

The algorithm iteratively fits regression functions f_{mj} to a response variable (reweighted residuals). One can build the f_{mj} by performing multiple regression based on all attributes present in the data, but it is also possible to use a simple linear regression, selecting the attribute that gives the smallest squared error. The optimal number of LogitBoost iterations M to perform is cross-validated, which leads to automatic attribute selection. This implies that the model learned after a few iterations will only include the most relevant attributes present in the data.

Therefore, when using the LLR algorithm in WEKA, the main parameters are the maximum number of iterations for LogitBoost, and either the iterations are cross-validated or the training set stopping criterion is used.

Fig. 2.4 displays a part of a LLR output model.

Algorithm 2 LogitBoost Algorithm (J classes)

Start with weights $w_{ij} = 1/n$, $i = 1, \dots, n$, $j = 1, \dots, J$, $F_j(x) = 0$ and $p_j(x) = 1/J \forall j$

Repeat for $m = 1, \dots, M$:

(a) Repeat for $j = 1, \dots, J$:

i. Compute working responses and weights in the j^{th} class

$$z_{ij} = \frac{y_{ij}^* - p_j(x_i)}{p_j(x_i)(1 - p_j(x_i))}$$

$$w_{ij} = p_j(x_i)(1 - p_j(x_i))$$

ii. Fit the function $f_{mj}(x)$ by a weighted least-squares regression of z_{ij} to x_i with weights w_{ij}

(b) Set $f_{mj}(x) \leftarrow \frac{J-1}{J}(f_{mj}(x) - \frac{1}{J} \sum_{k=1}^J f_{mk}(x))$, $F_j(x) \leftarrow F_j(x) + f_{mj}(x)$

(c) Update $p_j(x) = \frac{e^{F_j(x)}}{\sum_{k=1}^J e^{F_k(x)}}$

Output the classifier $\operatorname{argmax}_j F_j(x)$

```

=== Classifier model (full training set) ===
SimpleLogistic:
Class 5 :
-32.95 +
[minV] * -25.24 +
[Deltav] * -1.34 +
[typeroute=highway] * 26.73

```

Figure 2.4: Example of visualization of a part of the LLR output model.

2.2.6.3 K-nearest neighbors

This method will not be used during the data-mining process, but will be used to confirm the results. This method is also rather widely used in related works.

The algorithm implemented in WEKA is referred to as *IBk* and is based on [38]. *IBk* is an instance-based learning algorithm (IBL). IBL are algorithms which use specific instances rather than pre-compiled abstractions during prediction tasks. These algorithms can also describe probabilistic concepts because they use similarity functions to yield graded matches between instances. IBL algorithms are derived from the nearest neighbor pattern classifier [52]. They are also similar to edited nearest neighbor algorithms [53], [54], [55], which save and use only selected instances to generate classification predictions.

k-NN input consists of the k (a positive integer) closest training examples in the feature space. Thus k-NN needs to compute a distance between instances in order to estimate the closest neighbors. Then k-NN yields the class membership of the object. The object is classified by a majority vote of its k neighbors, the object being assigned to the most common class among the neighbors. If $k = 1$, then the object is assigned to the class of that single nearest neighbor.

One can select the most relevant distance function depending on the input

data: Euclidean, Manhattan, Chebyshev, Levenshtein and so forth. Here is a brief reminder of the distance formulas. Minkowski distance between two points $X = (x_1, \dots, x_n)$ and $Y = (y_1, \dots, y_n)$, where x_i and y_i are the value of the i^{th} (out of n) attribute for X and Y , respectively, is:

$$d(X, Y) = \left(\sum_{i=1}^n |x_i - y_i|^p \right)^{1/p} \quad (2.8)$$

The Manhattan distance is the Minkowski distance with $p = 1$:

$$d(X, Y) = \sum_{i=1}^n |x_i - y_i| \quad (2.9)$$

The Euclidean distance is the Minkowski distance with $p = 2$:

$$d(X, Y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2.10)$$

And The Chebyshev distance is the limiting case of the Minkowski distance when $p \rightarrow \infty$:

$$d(X, Y) = \max_{i=1, \dots, n} (|x_i - y_i|) \quad (2.11)$$

Another metric proposed by WEKA in *IBk* algorithm can be the Levenshtein distance. This is a string metric for measuring the difference between two sequences. The Levenshtein distance between two words is the minimum number of single-character edits (*i.e.*, insertions, deletions or substitutions) required to change one word into the other.

The primary output of IBL algorithms is a *concept description* (or *concept*). This is a function that maps instances to categories: given an instance drawn from the instance space, it yields a *classification*, which is the predicted value for this instance's category attribute. Thus, the IBL algorithms are described by three components:

1. **Similarity Function:** This computes the similarity between a training instance $X = (x_1, \dots, x_n)$ and the instances in the concept description. Similarities are numeric-valued and can be one of the distances previously presented.

2. **Classification Function:** This receives the similarity function's results and the classification performance records of the instances in the concept description. The function yields a classification for x .

3. **Concept Description Updater:** This maintains records on classification performance and decides which instances to include in the concept description. Inputs include x , the similarity results, the classification results, and a current concept description. It yields the modified concept description.

IB1, *i.e.*, IBk with 1 neighbor, is identical to the nearest neighbor algorithm except that it normalizes its attributes' ranges, processes instances incrementally, and has a simple policy for tolerating missing values.

The number of k nearest neighbors and the distance are the main parameters when using k -NN in WEKA. In related works, k is often below 10. We will test values from 1 to 10 with our data, with Euclidean and Chebyshev distances.

2.2.7 Summary on knowledge discovery with driving data

We presented in the previous paragraphs an overview of knowledge extraction from driving data, focusing especially on driving events and driving behavior. We also included in our research the driving parameters used in ADAS, the driving styles applied in simulation works, and more generally the studies concerning features impacting fuel consumption. Our goal was to provide a wide overview of the attributes having a major role in driving events and driving styles, as well as the different machine-learning methods and results used in related work. We see that driving data and driving events are often application-dependent, as are the driving features. We aim to recognize driving events which are not similar to those in the literature. This is because we propose to study the events defined by how the driver uses the actuators and we focus especially on longitudinal efforts to be able to implement our results on chassis-dynamometer in the latter stages of the work.

Even if many related studies focused on lateral accelerations, most common features concerning driving focus on speed, acceleration and environment, which is transferable to our work. Moreover these features are vehicle-independent parameters, *e.g.*, features independent of engine speeds, engaged gears *etc.* Finally, the machine-learning methods that we chose are widespread ones, similar to what related works propose. However, they are also specific to our need to understand the classification for the future implementation.

The process of our study is to recognize the events, then to characterize the driving behaviors on these events. Once we are able to describe the behaviors, we have to estimate which behavior is the most efficient for rational driving. And finally, we aim to simulate different driving behaviors in simulation and on chassis-dynamometer. To this end, we have to implement the driver models on our test facilities. Therefore, the following paragraphs focus on the state-of-the-art driver models and the methods of implementation.

2.3 Related work on driver models

2.3.1 Historical background

Driver model is a widespread term and can be used in many different fields covering a number of perspectives. Some studies will focus on modeling driver mental load, whereas others will be interested in the control of the steering wheel, and so forth. However, the historical background of driver modeling is common to all the objectives and was developed by Bellet in his study [56] concerning cognitive modeling and simulation applied to the activity of driving.

In this current work, we will not go into the definition of driving activity and its implications for the driver, *i.e.* the capture of information, the identification of the situation, the decision-making process, and finally the action. We suppose that our driver model is perfectly able to perform whatever it is asked to, and is only limited by the vehicle and the road. For example, we do not consider that a driver is not able to brake because of an excessive mental load.

Concerning the development of driver models, in the 1970s, the studies were focused on task analysis. Mc Knight and Adams [57] listed more than 45 tasks classified in 9 categories. Other studies [58] already proposed a hierarchical model of the driving task. They inspired other models based on three levels of performance: macro performance, situational performance, and micro performance.

In the 1980s, research works were oriented towards risk analysis with the objective to understand the traffic accidents causes. Several models (Wilde [59], Naatanen and Summala [60], Fuller [61]) proposed risk assessment studies, for example with a return-risk ratio and a target level of risk, or trying to have a zero-risk of collision. In the continuity of the hierarchical models, Van der Molen and Botticher [62] or Hollnagel [63] also emphasized on the hierarchical nature of driver decision-making. They describe the different processes (perception, judgment, decision-making) at the aforementioned three levels of the driving task. Rasmussen [64] [65] equally presents a model with a hierarchical conception of the driving task. The model is based upon three levels of control: skills level, rules level and knowledge level. These three levels are defined by task characteristic and no longer by an individual's internal processes. Michon [66] applies Rasmussen's model to driving task. In all these studies, authors point out the differences between strategic decisions (road planning), tactical decisions (maneuvering), and operational decisions (execution tasks). For our work, we focus on tactical decisions since the strategical tasks and the operational tasks are managed by the choice of the cycle and the controller, respectively.

In the 1990s, the first simulation models appeared thanks to I.T. development. Compared to the previous models, they are concrete and precise, as they are not only theoretical. The objectives are to reproduce or simulate the driving activity. In these models, Bellet differentiates the performance models, whose objective is to predict the driver behavior, and the cognitive models whose objective is to explain driver behavior. In the cognitive models, the driver is considered to be an information processing system. Driving behavior is not the central part of these models since they try to simulate cognitive system processes and its internal states when the system is in contact with environment.

We are more interested in the performance models whose objectives are to simulate the driver behavior and to obtain results comparable to real world. These models are based on relations between parameters, *i.e.*, contextual factors, and driving behavior. The inputs of these models are information coming from the environment, *e.g.*, the road and the vehicle, and the outputs are the performance built

by the model. The model itself is determined by technical considerations without considering cognitive behavior.

The first example is the Integrated Driver Model (IDM) by Levison [67]. Levison presents the development of a simulation model that makes driver behavior prediction possible. The model includes two subsystems: a procedural model and a driver/vehicle model. There is no distinction between a situation and its representation by the driver. Our study is based on the same hypothesis: the driver considers the right situation at the right time. The procedural model contains a scenario generator (*i.e.*, road, environment, situations), a simulation system for parallel driving tasks, and a cognitive model which allocates cognitive resources to the driver. Actually they do not really simulate driver behavior, but rather evaluate the impact of a side activity, *e.g.*, having a phone call or control of the radio, based on the response time of the driver.

Since then, performance models have been oriented towards traffic simulation alongside of risk analysis and safety studies. Driver models are limited to a control-loop: perception, decision, and action. Nowadays, more and more driver models are used for the development of new vehicles via simulation tools, and more rarely via chassis-dynamometers. We will consider that the perception of the road and the environment is not altered. Therefore, when the driver model receives the information stating that the driving event is a *toll*, or that the target speed is 90 km/h, for example, there is no differences between what the driver model perceives and what is provided by the road model. This is due to the fact that we focus on the behavior of drivers in a similar situation, not on how different drivers perceive the driving situation. Similarly, for the action, we will consider that the driver is not limited by any mental load, distraction, drowsiness, and so forth.

2.3.2 Adapt a driver model to different driving behaviors

We put forward three methods to model different driver behaviors in a control system that can be used together:

- Tune the control system parameters in order to obtain a different response to the same inputs. For example, putting different integral gains on a PID controller. However these parameters offer limited differences between drivers since they only have an impact on the precision, the rise time and the stability of the system.
- Tune driving parameters, *e.g.*, maximum acceleration, maximum braking and G-G diagram. They prevail over controller parameters because they are the limits of the outputs of the control system.
- Tune the setpoints, *e.g.*, speed, acceleration or steering wheel angle. The setpoints often depends on driving parameters, but are added to a preprocessing

which computes a different road for each driver type. For example, road pre-processing output can be a 90km/h speed target for a fast driver and a 80km/h speed target for a normal driver.

The following paragraphs present five examples illustrating these methods. We will then explain the method that we have chosen.

2.3.2.1 Tune the control system parameters

Data, Pascali, and Santi [68] propose a safety study concerning lateral behavior with steering wheel and emergency braking during accident phases. They do not discuss any specific dimensions of driving skill or style. They vary the preview time and control gain parameters of their model, resulting in variations in steering wheel behavior that are qualitatively similar to what have been observed for three drivers on a test track.

2.3.2.2 Tuned driving parameters and tuned setpoints: example of the Kiencke and Nielsen model

Kiencke and Nielsen [31] propose a complete hybrid driver model containing the following blocks: information acquisition, information processing, action control, perception, decision-making process, response execution, and resources management. The controller is a RST structure for both lateral and longitudinal dynamics. The following driver parameters are used to differentiate beginner, professional, and fast drivers:

Foresight time: t_f

Factor representing driver type: F

Potential braking acceleration: Ax_{pot}

Time span before entering the curve: t_{antic}

Lateral acceleration tolerated by driver: Ay_{pot}

Time to accelerate before end of curve: t_{acc}

The inputs of the lateral and longitudinal controllers are the acceleration reference values (Ax_{ref}, Ay_{ref}), information about the driving state and road course. Their outputs are gas pedal position (longitudinal), brake pedal force (longitudinal) and steering angle (lateral). Concerning the longitudinal model, the brake system and throttle system are previously identified with a test drive on a straight dry road and providing *Torque drive* and *Torque brake* maps. Then nonlinear maps are built to produce the torque $Torque_{ref}$ table depending on acceleration and speed ($Ax, Speed$). Finally, two RST structures (in a GPC controller) provide brake and throttle models. These models cannot be active at the same time.

Simulation results, shown in Fig. 2.5, illustrate that the differences between drivers on the longitudinal axis are mainly explained by the speed target. This is an interesting method since it is implemented in simulation and provides different driving behaviors, however the driving behaviors are not applied to rational driving or fuel savings.

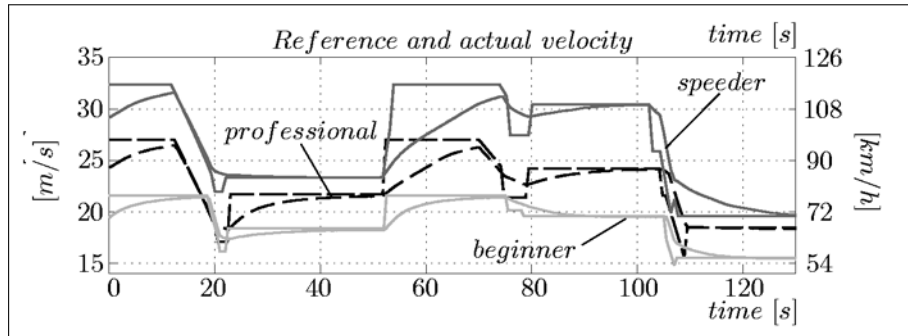


Figure 2.5: Simulation results for three driver types with Kiencke and Nielsen driver models.

2.3.2.3 Tuned driving parameters and tuned setpoints: example of the IPG driver model

IPG Automotive¹ [69] offers a Professional Simulation Environment for Virtual Test Driving, including Vehicle-in-the-loop tests. Their driver model is called *IPG Driver*. Driver model inputs are only information that can be recognized by a real driver: position, velocity, acceleration, yaw angle, engine speed, gear number and clutch position (and steering wheel torque optionally). In addition to these signals, the outputs are steering wheel angle, accelerator pedal position, force on brake pedal, position of the clutch pedal, and gear number. Each parameter of the driver model impacts the way it will drive. For example, the user has to change some parameters to simulate a race driver instead of a normal driver: higher G-G exponents (see paragraph below), driving with maximum speed enabled, shifting and pedal switching time reduced, corner cutting coefficient, and knowledge turned on.

The G-G diagram is a diagram with the longitudinal acceleration a_x on the vertical axis and the lateral acceleration a_y on the horizontal one. This diagram is often used in racing sports to observe how the driver is driving in a curve or out of a curve. In the latest version of IPG driver's, the driver's first task is to follow the preceding car, *i.e.*, following a speed reference.

The definition of different driving styles depends on allowed minimum and maximum distance (with the preceding car), minimum and maximum time gap, Tolerated Speed Deviation / Detection Threshold, and an EcoCoefficient to vary the driving

¹<http://ipg.de/company>

style. The IPG Driver model (v6.0) uses the following control strategy for path tracking and speed control.

The general idea is to divide the controllers' work into two parts. First, before starting the simulation, everything that is not changed during the simulation is calculated. This means that when the driver model is first used, a statically desired course is calculated from the road information (*e.g.*, the corner cutting coefficient). Staying on this static desired course is the goal of the steering controller.

The static desired course is also the input for the choice of the speed. With the information about cruising speed and maximal longitudinal and lateral accelerations, the static desired velocity is calculated along the length of the course. Keeping the driving speed close to this static desired velocity profile is the goal of the longitudinal controller. This first part creates the path of the course (curves, straight lines) and builds the reference speed all along the length of the course.

The second part is the controlling task during the simulation. To get a realistic control behavior, a prognosis of vehicle movement is made. The driver has to know the dynamic behavior of the car in order to react correctly to the prediction. This is made in general with a learning procedure, where, *e.g.*, a yaw response time is identified. The information that is learned in the learning procedure is called the driver's knowledge and is stored in a knowledge file. The user has to activate the learning controller in order to learn the vehicle's limits. Therefore, the complete configuration required for the tests is recorded by the system in the knowledge file during the learning phase. This file contains the results of the learning maneuvers which can be presented in tables or maps containing information such as (*speed, a_x, brake, a_y*) or *max speed*.

Then the simulation can start using this knowledge, even if some parameters may need to be adjusted (such as G-G exponents to adapt to higher speeds). During a simulation, an identification of the current dynamic situation is carried out as well. With this information, the driver adapts the control behavior in order to calculate the correct feedback signals in steering and accelerating, or braking in each situation.

We were able to test this model with IPG Driver software. IPG driver (with different configurations) has globally an oscillating behavior. Although oscillations may represent some real driving cases, the model does have low performance results from a regulation point of view (high overshoots, oscillating commands). This oscillating problem also plays a role in the controller operational speed. In a braking phase, the GSP driver can be faster than IPG driver: because even if GSP driver brakes less, it stabilizes more quickly at the target speed, as we can see in Fig. 2.7. IPG driver does not make use of coasting. Brake and throttle pedals are continuously used.

Table 2.6: Example of driver parameters to differentiate normal and racing driver in IPG driver software

Parameter	Normal driver	Race driver
Duration of accel.-brake change [s]	0.5	0.1
G-G - Diagram: Exponent for acceleration	1.0	2.0
G-G - Diagram: Exponent for braking	1.0	1.5
Driving with max. speed (1=yes, 0=no)	0	1
Corner cutting coefficient[0...1]	0.8	1

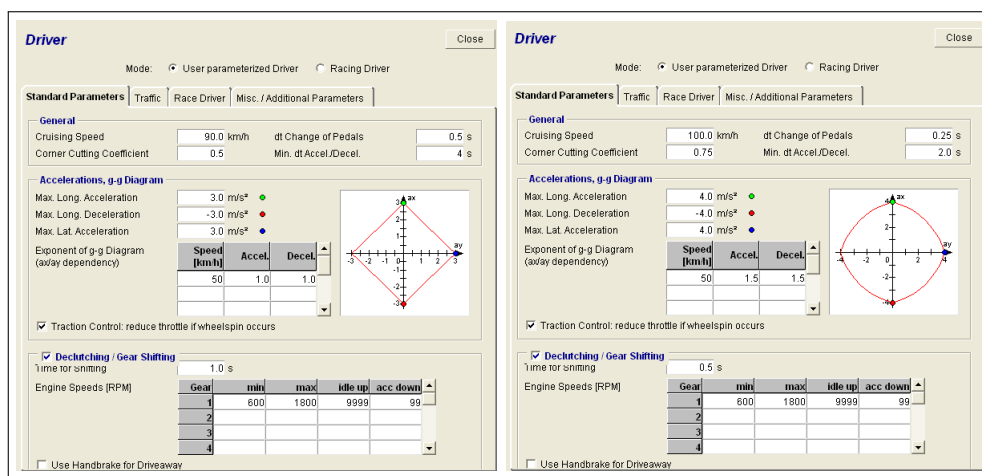


Figure 2.6: IPG driver parameters for aggressive (on the left) and normal (on the right) drivers.

2.3.2.4 Example of a driver model including driving events: COSMODRIVE model

The COSMODRIVE model [56] [1] [70] and [2] is a model of cognitive simulation of driving activity developed at INRETS-LESCOT. COSMODRIVE means COgnitive Simulation Model of the DRIVER. This model is based on data collected real world data, *i.e.*, real vehicle and human drivers. We studied this model in details because the methodology can be similar to ours.

The general architecture of COSMODRIVE model is presented in Fig. 2.8. Authors distinguish three levels within the model: strategic, tactical and operational blocks. The *strategic* module simulates processes of route-planning and navigation. The *tactical* module generates internal representations of the road environment and makes decisions. Knowledge and categorization mechanisms are included in the *tactical* module. This module anticipates several situations from the current situation. The prognoses are then fed into the operational module.

The *operational* module is a set of operational units which represents implementation of elementary tasks (lateral control, speed control, distance control). The

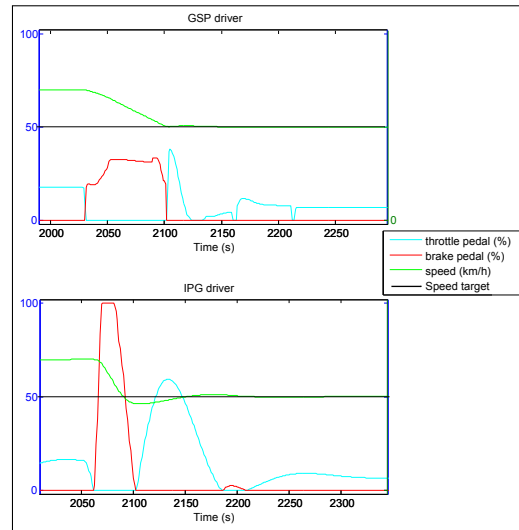


Figure 2.7: GSP driver behavior (top graph) and IPG driver behavior (bottom graph) during a braking phase.

operational module coordinates these units to supply the right sequence order to the execution module. There is also a specific management module to handle emergencies.

Finally three additional modules represent the link with the road-environment and the management of cognitive resources. The *execution* module is the interaction with the vehicle and carries out elementary tasks (steering wheel, pedals command).

The *perception* module is the interface between the model and the environment.

And the *management and control* module manages resources between the various cognitive processes, similar to an operating system.

The *tactical* and *operational* modules are the most interesting for our study because we consider that the driver is able to understand a situation and to execute appropriate commands in an optimum way.

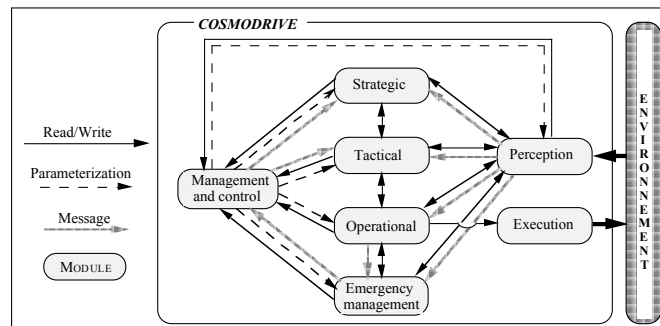


Figure 2.8: General architecture of COSMODRIVE [1]

In the COSMODRIVE model, driving situations including crossroads, roundabouts, straight road and turns are considered in different road environments. In the tactical module, the categorization and recognition of driving situations is done with *driving schemata* representing the knowledge of each driver about each situation. They include the tactical goal of a situation (*e.g.*, left-turn), the infrastructure, the sequence of actions, and states (*e.g.*, speed and position in different zones to reach the target). Fig. 2.9 presents an example of these driving schemes. Each schema translates the knowledge of a situation into appropriate actions. For example, their database of 626 left-turn situations have been separated into 2 strategies (with, or without traffic) and the strategy without traffic has also been separated in 3 sub-strategies.

The tactical module is composed of three cognitive agents: blackboards describing the working memory, knowledge (of the driver) and processes (modeling the mental activities). In order to implement this theoretical model, Bellet *et al.* [70] interfaced COSMODRIVE and a vehicle-environment platform called SiVIC. This platform was developed by INRETS-LIVIS [71] for virtual design of driving assistance systems. Finally this model is designed to simulate drivers' mental representation and is consequently limited concerning the behavioral simulations that we intend to do. Moreover, the behavioral performances resulting in driving performances are designed to safety purpose which is different from our concern on rational driving. However, an interesting point is the adaptation of their theoretical model to an existing simulation tool. They had to adapt their model in order to connect the two existing tools. In our study, we will concentrate our efforts on directly implementable solutions.

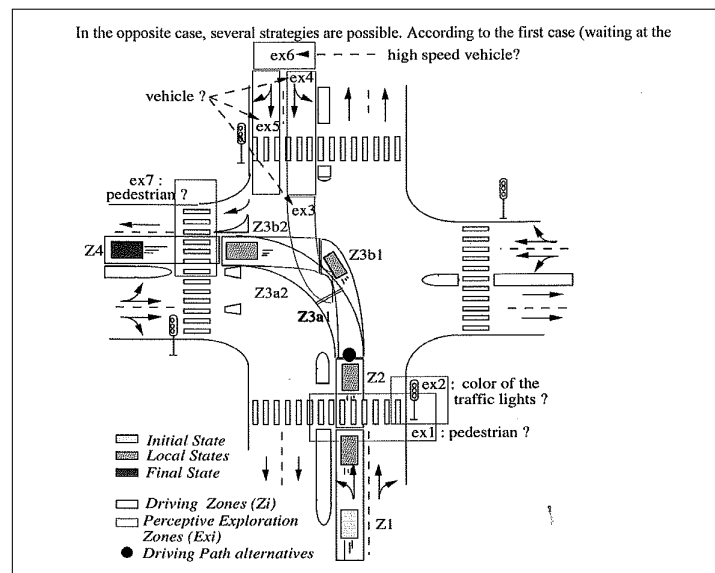


Figure 2.9: An example of COSMODRIVE driving schema: left turn at an urban crossroads [2]

2.3.2.5 Example of a truck driver model: ACEA model

In order to have more comparable figures between truck manufacturers and to improve development processes, the ACEA European workgroup *ACEA Workgroup CO₂-HDV* is currently working on a topic which aims to select an approach for a certified CO₂ declaration in the whole European Union, based on 3 general statements [72].

- CO₂ and fuel efficiency values have to be realistic for all manufacturers, to be considered vehicle variants, and consequently ensure accurate calculation of CO₂ saving in real world.
- As far as possible, CO₂ saving measures should be covered, but the declaration procedure needs to be repeatable, robust, and practicable.
- The effort and resources required for CO₂ declaration need to be reasonable for OEMs².

Given that only a full vehicle approach is able to fulfill these premises, ACEA workgroup targets a full vehicle approach with a global simulation tool, including cycles, driver model, and vehicle configurations. The final goal is that customers will have access to official fuel declarations of standard vehicle configurations.

Their cycles are defined as altitude and slope profiles *vs.* distance, and desired speed *vs.* distance. Stops and stop time have to be included. There will be a cycle for each usage (long haul, regional delivery, urban delivery, municipal utility and construction). Their specification for the driver model is to have only one driver model, but different generic parameters should be set to cover different driver behaviors in specific vehicle applications and to handle manual transmissions and automated manual transmissions too.

An accelerator pedal actuation and shift logic depending on defined maximum desired acceleration is needed in order to obtain a realistic driver model behavior. These desired acceleration curves have to be vehicle class specific (different parameter sets within one driver model) as shown in Fig. 2.10. This leads to full throttle acceleration in most cases, except for some vehicles with very high performance or with specific conditions like refuse vehicles or city buses.

There are plans for an advanced driver model which will manage uphill under-speed with delayed return to normal set-speed, downhill over-speed, free rolling with over and under speed limits. The proposed driver model is different to others that can be found in the literature since the objective is to actually implement it. However, this model is not based on an analysis from real driving behavior data.

²Original Equipment Manufacturers

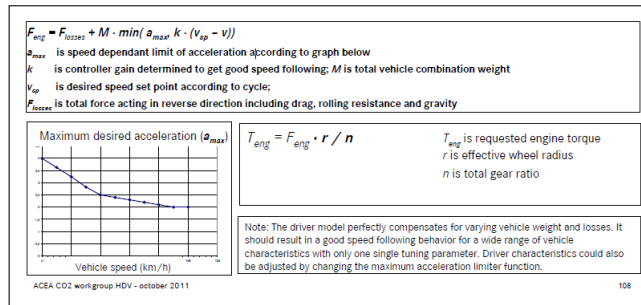


Figure 2.10: Example of acceleration strategy implementation in the ACEA driver model.

2.3.2.6 Summary

We have seen in the previous paragraphs several examples of driver model specification and implementation using different methods. Some methods tune the control system parameters to obtain a different response to the same inputs. Other methods tune driving parameters, and some tune the setpoints, which imply different inputs to the lateral and/or longitudinal controllers depending on the desired driver behavior. However, these works are not directly comparable with our study, because their purpose is different, *e.g.*, the COSMODRIVE model focused on mental load and road safety, or because they are not always based on an analysis of real driving data, *e.g.*, the ACEA model, IPG model. The challenges in our work are to create driver models based on real driving data, but implementable in our exiting tools and rational-driving centered.

In the next Chapter, we will present our data collection and the automatic recognition of driving events on real data and on simulated data.

Driving events recognition

As stated in the previous chapters, automatic driving events recognition along with driver behavior classification is essential for accurate estimations of fuel consumption and average speed during the truck development process. In Chapter 1 we presented our method, which is the recognition of driving events, then the identification of driving behaviors parameter when approaching some driving events. The Chapter 2 exposed the KDD process and the main parts of the process, *i.e.*, data, feature selection and methods, applied to the related studies concerning driving data analysis. This chapter focus on the automatic recognition of driving events, and we especially detail the data collection, the driving events definition, the selection of driving features, and the results.

First, we propose a learning-based classification method to recognize driving events from real driving data. Then we apply the learnt model to digital roads. These simulated road data are mostly synthesized from real driving data. Since we aim to reproduce driving behaviors in various driving events in a simulation environment and on a chassis-dynamometer, the events recognized with real driving data must also be recognized on digital road signals.

For the purpose of driving events automatic recognition, we define the set of significant driving events in which drivers can display different behaviors. We also introduce a set of features potentially relevant to discriminate these events. Finally, we use of two machine learning methods presented in Chapter 2, namely decision tree and linear regression models, to classify these driving events.

The contributions are fivefold:

- Identification and categorization of road events impacting driver behavior;
- Proposition and analysis of a set of features related to small-scale driving patterns sensitive to road events;
- The use of two machine learning techniques, namely decision trees and linear logistic regression, for automatic recognition of road events from small-scale driving patterns;
- The identification and the classification of driving events with real driving data. It includes the selection of the main discriminating features for driving events classification.
- The experiments of the learnt models carried out on both real driving data and digital roads that are simulated driving data.

3.1 Driving events definition

We have seen before that there are many ways to define driving events according to the objective of the underlying study. In this work, our objective is to understand the driver behavior and we are interested in driving events in which drivers can display different behaviors. To control the vehicle speed when facing an event, a driver will use actuators, namely *throttle pedal*, *brake pedal*, *retarders* and *coasting*. They are indicated with arrows in figure 3.1. Throttle and brake pedals are the well-known pedals used to accelerate and brake the vehicle respectively. Retarders are devices helping the functions of primary friction-based braking systems. *Coasting* is when the driver uses none of the previous three actuators. In this study, driving events are delimited according to the use of these actuators and result in two different categories. The first category includes *throttle>0* events starting each time the driver starts using throttle pedal. Conversely we observe a *throttle=0* event in the second category when the driver uses the brake and/or retarders in between two adjacent coasting periods. Figure 3.2 shows an example of these two event types. However, straightforward application of the previous definitions results in a number of driving events far higher than the reality. As a result, four more rules are defined with respect to the way a driver controls his vehicle depending on the road situation:

- Obviously there is no event when the vehicle is not moving (speed equal to 0km/h).
- If two *throttle>0* events are separated by a coasting period and the speed gap is less than 3 km/h between them, these two events are merged. Indeed if the driver lets the vehicle gain or lose a significant speed while coasting, it means that he wants to change his target speed so it should be interpreted as a new event.
- If two or more *throttle=0* events are too close to each other they are mixed (*i.e.* $\Delta t < 5s$ between the two events and engine torque must be < 15 N.m during this period, as well as percentage of throttle position must be $< 5\%$).
- If the coasting phases of two adjacent *throttle=0* events overlap each other, these events are also merged into a single one.

In the following subsections, we detail the various driving events occurring in *throttle=0* event class (subsection A) and *throttle>0* event class (subsection B).

3.1.1 Throttle=0 events

During the data acquisition (explained in 3.2) we observe 16 causes, presented in Table 3.1 and Fig. 3.3, that could involve the use of the brake and/or retarders. These causes are access ramp (*i.e.* braking to move into a line of traffic), give way, speed bumps, red traffic light, truck size constraint (*e.g.* narrow road), low speed (*i.e.* start-stop events occurring in a traffic situation where speed does not

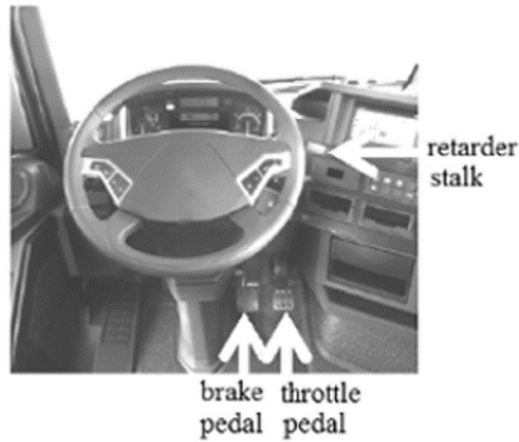


Figure 3.1: Actuators that the driver can use to control the speed of the truck.

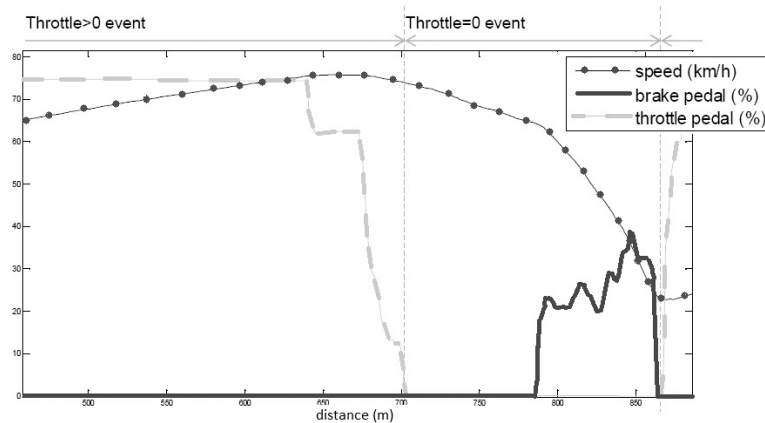


Figure 3.2: Illustration of the two types of driving events: a $throttle > 0$ event followed by a $throttle = 0$ event. In the first case, throttle pedal is in use whereas brake pedal is engaged in the second case

exceed 10km/h), lower speed limit, toll, pedestrian crossing, keeping constant speed downhill (*i.e.* because of the kinetic energy of the vehicle, the driver must brake in steep slopes in order not to excessively overspeed), roundabout, exit ramp, stop sign, traffic (*e.g.* traffic jam still in movement, unexpected car moving into the traffic), roadwork, and curve. However, access ramps, roadworks, and pedestrian crossings are events, which, with less than 1% of all events, rarely occurred in our dataset. They are deemed insufficient for a reliable machine-based learning. As a result, each time it is relevant, some of these occurrences are labeled as *keep constant speed downhill*, *truck size constraint* or *low speed* according to the video recording. Therefore at the beginning of the study there were 13 classes for $Throttle=0$ events.



Figure 3.3: Examples of driving events: speed-bump, red traffic light, curve, give way, roundabout, lower speed limit, truck size constraint (2 examples), keep constant speed in downhill, traffic, toll, access ramp, stop and low speed, respectively.

3.1.2 Throttle>0 events

Concerning the *Throttle>0* events, we observe 6 events that could cause the use of the throttle pedal and presented in Table 3.2. These events are: driving at a speed limit, reaching a higher speed limit, urban take-off, main-road or highway take-off, adjusting to stop, and adjusting for other reasons. The last two classes are

Table 3.1: *Throttle=0* events definition

Start Event E_i at time t	$ThrottlePosition(t - 1) > 0$ AND $ThrottlePosition(t) = 0$
End Event E_i at time t	$ThrottlePosition(t - 1) = 0$ AND $ThrottlePosition(t) > 0$
Events	A: roundabout, B: traffic, C: speed bumps, D: red traffic light, E: truck size constraint, F: curve, G: keeping constant speed in downhill, H: lower speed limit, I: stop sign, J: toll, K: low speed, L: give way, M: exit ramp.

different from *driving at a speed limit* because they are much shorter. They only include events when, because of a changing situation or because the driver does not anticipate the distances well, the driver presses for a short duration ($< 2s$) the throttle pedal then releases it in order to adjust the vehicle speed.

We also build a detailed list of all the driving situations with expert drivers, presented in Appendix B. It appears that the classes identified during our data acquisition cover the main and the most common driving situations.

Table 3.2: *Throttle>0* events definition

Start Event E_i at time t	$ThrottlePosition(t - 1) = 0$ AND $ThrottlePosition(t) > 0$
End Event E_i at time t	$ThrottlePosition(t - 1) > 0$ AND $ThrottlePosition(t) = 0$
Events	A: driving at a speed limit, B: reach a higher speed limit, C: urban take-off, D: main-road or highway take-off, E: adjust for stop, F: adjust for other reasons.

3.2 Driving data acquisition

Machine-based learning requires representative training data for both learning and validation of a learnt model. Unfortunately, we did not have any driving data suitable for learning when we started this study. Therefore, we launched a process of data acquisition with the objective of covering the maximum number of driving situations possible. We made two data collections (in 2011 and 2012, respectively) for the purpose of driving events recognition. A third data collection was carried out in 2013 for the purpose of driver behavior analysis and will be presented in Chapter 4. For now, we will present the two first data collections.

3.2.1 Data collection for the recognition of Throttle=0 events

We chose a vehicle and several roads representative of a regional distribution use. It means that the routes took place in urban and extra-urban areas with a few highway roads. All the data were acquired in the Lyon region, France. The vehicle was a VOLVO FL as illustrated in figure 3.4, with a robotized gearbox, equipped with two sets of engine retarders (exhaust and compression) and loaded to 13.7 tons.

The vehicle was equipped with the devices presented in Fig. 3.5 for the purpose of data collection. These devices are a CAN data logger recording CAN bus signals such as vehicle speed, throttle pedal percentage, *etc.*, a GPS for vehicle geolocation, and a camera recorder for filming the outside scene. Only CAN data and GPS data are used for classification. The camera recording helped label the events as well as allow understanding of what is actually happening on the road during events, so as to explain classification results. Indeed, the collected raw driving data then need to be further manually post-processed to segment various driving events used as ground truth data in the latter learning stage. However, without images showing the traffic environment, this manual segmentation proved to be very complicated. As a result, we implemented a specific annotation tool, which shows both the video on the traffic environment and driving data in a synchronized way.

During data collection, drivers were asked to drive normally but without using cruise control, and to always keep the robotized gearbox in auto mode. We laid down these rules to separate the impact of the driver from the vehicle. Since driving events are analyzed partly via the actuators (see 3.1), it was important that the driver could control vehicle speed only with the actuators and not by using the cruise control. For the same reason, we did not want the driver to control the gearbox, so we asked him to keep it in automatic mode.

Fig. 3.4 sketches the process from data collection to the driving events database. The raw driving data includes several hundred kilometers of data from different trips with the same vehicle. In each of these trips, we need to extract the different driving events. This database represents 19 hours of driving over 952km with 913 *throttle = 0* events. *Throttle > 0* events were not studied in this database. We will refer to the data coming from this data collection as **Database 1**.



Figure 3.4: FL vehicle

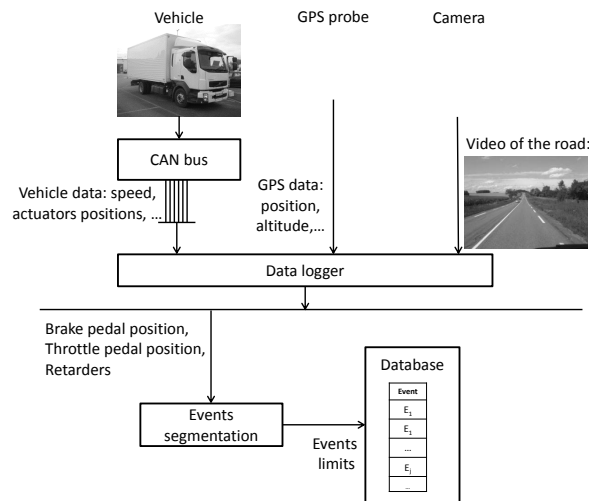


Figure 3.5: This figure represents the stages from the data collection to the driving events. A data logger acquires signals from CAN bus, GPS and camera. Once all the data is collected, we segment the driving data into driving events. After that, we can calculate the features values, and use the database to perform the classification.

3.2.2 Data collection for previous work validation and for the recognition of $Throttle > 0$ events

In order to validate the work done with the first database, a new acquisition campaign was carried out. The goal is to have validation data and also new data for $throttle > 0$ events. Two vehicles were used:

- The same VOLVO FL vehicle as on the first data collection, loaded to 13.7 tons.
- And a RENAULT TRUCKS Premium to illustrate long haul usage, loaded to 33 tons. This vehicle is presented in Fig. 3.6. The data coming from this vehicle will be used to validate the models learnt with previous data collection on $Throttle=0$ events recognition.



Figure 3.6: RENAULT TRUCKS Premium

Data were recorded with a GPS, a data logger for CAN data, along with a synchronized video recorded throughout campaign duration. We also made a switch box illustrated in Fig. 3.7. During the acquisition, the accompanying person pushed the different switches when there was a $Throttle=0$ event and the data logger recorded the appropriate voltage. This box avoids manual labeling since each voltage is equivalent to a different event.



Figure 3.7: Box events

Two representative cycles were selected: one long haul usage (148 km) and one distribution usage (252 km) around Lyon once again. These two cycles were chosen amongst the other cycles run during the first data collection for two reasons: they have a similar percentage of each class of event as the whole initial database, and classification for $Throttle=0$ events with these cycles is coherent with the global classification rate. The new database must assess driver impact so 11 drivers drove on the distribution cycle with the FL truck, including 2 drivers from the Driving Performance team and 9 drivers working temporarily in our proving ground (LVPG). To estimate vehicle impact, our reference driver from the first data collection drove

the two reference cycles with the two vehicles.

We will refer to the data coming from this data collection as **Database 2**.

3.3 Driving events sensitive features

Once all the driving events had been identified and manually labeled on the data that was acquired during the acquisition process, we had to extract meaningful features that are sensitive to these driving events for latter stage of machine learning.

As listed in Table 3.4, we collected and divided a set of potentially sensitive driving events features into three groups. These three groups are defined by the main components impacting the speed and the fuel consumption of a vehicle, thus helping to characterize the various aspects of a driving event. The first group of features relates to vehicle behavior (speed, acceleration *etc.*), whereas the second group relates to driver behavior (brake, throttle *etc.*). Finally, the last group contains the features relating to the environment (road type, slope *etc.*).

Most of these features already appear in literature, in particular those related to speed and acceleration. We adapt some of the features describing the subparts of an event with even driving or, on the contrary, with sudden acceleration. However, we modify the thresholds to suit the dynamics of a heavy vehicle. For example, the acceleration threshold for high acceleration for a truck is $|a| > 0.7m/s^2$ instead of $|a| > 2.5m/s^2$. We also add some other features to describe more precisely a driving event and its context, *e.g.*, duration and distance of an event which provide an overall picture of the event while slope, road type, and the remainder of the context features depict the environment. These features are based on a straightforward intuition that an event may depend on its context, *e.g.*, a toll is more likely to be expected on a highway road than on an urban road. Concerning road type, three categories are identified: urban road, main road and highway, as detailed in Table 3.3.

Table 3.3: Road type parameters

Road type	Characteristics
Highway	Highway with a 80 or 90km/h speed limit, At least two lanes in each direction, separated by a median.
Main road	Main road with a 70 or 80km/h speed limit, One lane in each direction, not separated.
Urban road	Urban road with a 30 or 50km/h speed limit.

We also introduce some distance features. They provide the measure the distance from the next/previous event of the same category (*i.e.*, $throttle > 0$ or $throttle = 0$). The reason is that we observed in our collected driving data that, in some areas, there

are sequences with many braking events whereas, in other places, there are far fewer. Distance from the next/previous event can thus be an indicator for the environment. Finally, we also compute various features concerning the driver's actuators use, particularly the percentage of time and distance from the event where the driver uses the brake, throttle or retarders. Intuitively, faced with different driving events, drivers may use the actuators differently. These features may thus provide clues as to the underlying driving events.

Table 3.4: Driving features for events classification

Group	Driving features
Speed and acceleration of the vehicle	Maximum, minimum, average and standard deviation of speed Δ speed: Gap between initial and final speed Speed gap at 75%, 50% and 25% of the event Maximum, minimum, average and standard deviation of the acceleration Percentage of even driving time ($ a < 0.28m/s^2$) Percentage of time when absolute value of acceleration exceeds $0.7m/s^2$.
Actuators use (driver)	Percentage of time when service brakes are used, maximum % of the brake pedal Percentage of time when throttle is used, maximum % of the throttle pedal Percentage of time when retarders are used Percentage of time when there is neither retarder nor service brake Percentage of time when both the retarders and the service brake are engaged Energy dissipated during braking and retardation phases Maximum and average % of engine torque Fuel consumption.
Environment	Event distance Event duration If this event finishes with a stop Previous and next event Distance gap from previous/next event Time gap with previous/next event Average speed within the last 1/3/5 km Stop density within the last 1/3/5 km Average slope Road type (urban, main road, highway).

3.3.1 Throttle=0 events

As the number of features listed in Table 3.4 is quite high and some of them may be intercorrelated, a key issue is thus to select a subset of these features that are sufficiently discriminating to distinguish the driving events identified in the previous section. In the field of machine learning, feature selection [73] aims to exclude redundant or irrelevant features for improved model interpretability, shorter training times and enhanced generalization skills of the learnt model by reducing overfitting. In this work, we use *Sequential Forward Selection* principle (SFS) [74] [75], which consists of testing growing subsets of features using classification accuracy as the stop condition. For this purpose, **the collected driving dataset as defined in section 3.2** is used as hard data.

In order to evaluate the most discriminating features to recognize *throttle=0 events*, SFS is applied to select growing subsets of features by adding a new feature at each iteration. For each subset we classified driving events using C4.5 and the linear logistic regression algorithm, and selected the subset of features which improves the classification rate. The toolbox WEKA used for SFS outputs the list of most relevant features. As a result, the following features appear to be the most important for classification: maximum speed, minimum speed, speed standard deviation, the gap between initial and final speed (Δ speed), road type, distance, stop, previous event, and distance with the next event. The experimental results based on growing subsets of features confirmed a well-known phenomenon, *i.e.*, utilization of more features may lead to a recognition accuracy decrease, since some are redundant and generate noise. For example, average speed in the previous kilometers does not provide more information than road type.

Once these 9 attributes have emerged, we decide to make a further manual choice between them. Our target is to obtain the best possible compromise between the number of attributes, the simplicity of calculating these attributes on real and digital data, and the classification results. For example, the classification rate (see Section 3.4.1 for classification details) is 73.5% with the following 9 features (maximum speed, minimum speed, (Δ speed), road type, distance, stop, speed standard deviation, previous event, distance with the next event), while the events are correctly found at 74% with only the following 6 features (maximum speed, minimum speed, (Δ speed), road type, distance, and stop).

All the results for *throttle=0 events* presented in the subsequent sections are obtained using these 6 features. Intuitively, the features selected by SFS are consistent with the expectations, *i.e.*, driving events can mostly be characterized by speed-related and context-related features.

3.3.2 Throttle>0 events

With respect to *Throttle=0* events work, we tested common features and particularly those relating to speed and acceleration. Since fuel is injected when the driver

engages the throttle pedal, we also introduce some features focused on fuel consumption, especially those derived from [8], such as the duration of evenly driving and the percentage of duration of sudden speed change. Furthermore, we proposed some new features such as context features (next event, previous event...). Table 3.4 lists all the driving features exploited for the automatic recognition of driving events.

Once more, SFS is applied to select the most discriminant features to recognize *Throttle=0* events along with C4.5, and linear logistic regression. As a result, we observe that the common attributes still play the major role, particularly Δ speed, minimum speed, average speed, speed standard deviation. However, unlike *Throttle=0* events, acceleration attributes have more impact. Fuel consumption also appears as a discriminating attribute.

Using selection of features, the best classification accuracies are obtained with the following subset of eight features: Δ speed, minimum speed, speed standard deviation, maximum acceleration, duration of sudden speed change, mean percentage torque, fuel consumption, and road type. In the same way as *Throttle=0* events, this result proves to be consistent with the features that we expect, with knowledge of drivers' behavior during these events. For instance, to discriminate between the event of urban take-off and that of main road take-off, given the fact that final speed should be less than 50km/h for urban take-off, while final speed for main road take-off is around 70-80km/h, the feature final speed is intuitively expected as a discriminating feature to classify these two events. It is worth noting, particularly for the sequel of our work, that we observe merely one percent in performance drop when only the following four attributes are used, namely Δ speed, minimum speed, speed standard deviation, and road type.

3.4 Driving events recognition from real driving data using two machine-learning algorithms

3.4.1 Classification of throttle=0 events

The **Database 1** contains 913 *Throttle=0* events divided into 13 classes presented in 3.1.1. Unless indicated otherwise all the results presented below are obtained with this database, and the C4.5 algorithm using 10-folds cross validation method. The dataset is thus partitioned into 10 folds of 91 or 92 instances each. Consequently the learning of the two former algorithms is carried out using 9 partitions while the learnt model is tested with the remaining partition. This is then repeated 10 times but each time using a different learning and testing set. However, it turns out that brute-force training of the two learning algorithms on the initial raw database for the 13 classes leads to a disappointing 49% classification rate, although this classification rate is already far better than a random classification. Some classes, *e.g.* toll, are well classified as can be seen in table 3.5, whereas others have a fairly low classification rate, *e.g.* curves.

Table 3.5: Classification rate with initial database

Classification rate (%)														
Class	All	A	B	C	D	E	F	G	H	I	J	K	L	M
TP rate	49	60	30	43	77	19	21	68	39	48	83	89	11	30
Instances	913	146	127	60	129	47	95	37	62	85	18	56	28	23

We remind you that our goal is to identify the driving events leading to different driver behaviors. We thus discuss these results with internal expert drivers, who have analyzed their driving approach via utilization of actuators. They found that they have similar behaviors with respect to several driving events that are seriously confused by the learning algorithms. In order to quantify such a similarity and improve recognition results, we measure the Fisher linear discriminant between each pair of classes, according to equation 3.1.

$$J_{1/2} = \frac{1}{n} \times \sum_{i=1}^n \frac{(\mu_{1,n} - \mu_{2,n})^2}{\sigma_{1,n}^2 + \sigma_{2,n}^2} \quad (3.1)$$

where $J_{1/2}$ is the Fisher distance between class 1 and 2, n is the number of features, $\mu_{C,n}$ is the average value of feature n in class C and $\sigma_{C,n}$ is the standard deviation value of feature n in class C .

This discriminant can be interpreted as a distance and is used to quantify inter-class similarities versus intra-class dissimilarities. Two examples illustrate this measurement in figure 3.8. *J: toll* class is quite far from all the other classes (on the left of the figure), whereas *B: traffic* class is close to a lot of classes. That is to say that this class is too close to many other classes to be distinguished easily.

Table 3.6: Fischer distance between classes

	Size constraint	Lower speed limit	Toll	Keep constant speed in downhill	Roundabout	Exit ramp	Stop	Traffic	Low speed
Size constraint	0	1.87	4.54	2.14	0.89	5.37	0.93	0.38	17.8
Lower speed limit	1.87	0	3.38	94.1	1.01	79.6	2.37	0.66	20.6
Toll	4.54	3.38	0	6.78	2.83	3.11	3.12	2.77	12.1
Keep constant speed in downhill	2.14	94.1	6.78	0	1.98	244	3.04	0.93	16.4
Roundabout	0.89	1.01	2.83	1.98	0	1.36	1.46	0.28	12.5
Exit ramp	5.37	79.6	3.11	244	1.36	0	4.01	1.19	50.5
Stop	0.93	2.37	3.12	3.04	1.46	4.01	0	0.53	7.21
Traffic	0.38	0.66	2.77	0.93	0.28	1.19	0.53	0	11.4
Low speed	17.8	20.6	12.1	16.4	12.5	50.5	7.21	11.4	0

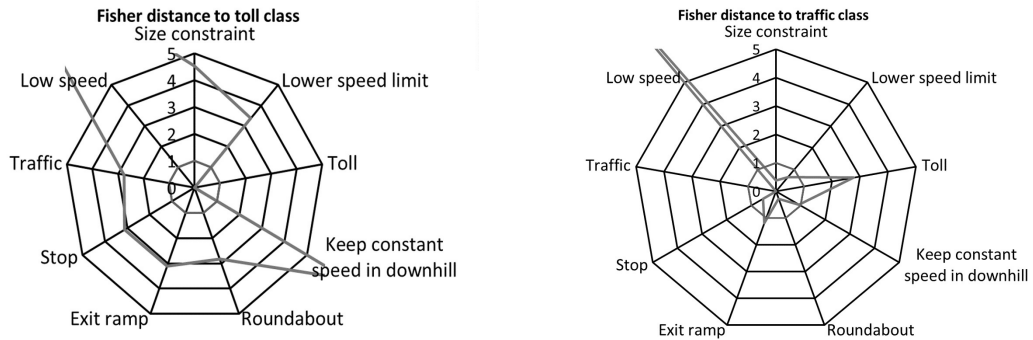


Figure 3.8: Illustration of the Fisher distance between classes. On the left, distance to toll class. On the right, distance to traffic class.

Expert drivers' advice combined with analysis of the Fisher discriminant lead us to modify labeling of the initial database with respect to driver behavior, and to group and/or separate some classes and instances. Each step is presented in the following paragraphs.

On close observation of speed graphs for class *E: size constraint*, we see that 5 instances needed to be switched to another class. It then appears that for class *F: curve*, high speed instances have a similar speed shape to class *H: speed limit*. These first changes increase the classification rate to 57%, especially for *H: speed limit* class (+18%) and *A: roundabout class* (+10%).

Further to the above merge of classes, we also find that an exit ramp event ending with a stop is quite similar to a toll event in terms of driver behavior. As a result, we move the instances of class *M: exit ramp* with stop to class *J: toll*.

Similarly, a stop event of class *D: traffic light* is quite similar to a stop event of class *I: stop sign*. This confusion can be observed in the confusion matrix shown in Table 3.7. The gray cells represent how these two classes are mixed up during the classification. When these two classes were merged, the classification rate rises to 63%.

We then study the *L: give-way* class, whose classification rate is still quite low. When examining the classification results, it turns out that instances with stop are always confused with class *I: stop*. As a result, we move these instances to class *I: stop*. Furthermore, we also merge the remaining instances of class *L: give way* with class *A: roundabout* as roundabouts are a variant of give-way. These changes enable us to reduce the number of classes and improve the classification result.

At this point, the classification results and the confusion matrix show that *C: speed bump*, *E: size constraint* and *F: curve*, are often confused. Their similarities can be observed via the speed curves of these events. These three classes are all grouped in a broader class referred to as *E: urban speed reduction*. This class groups all speed reduction events in urban areas except those caused by traffic or

Table 3.7: Confusion matrix

A	B	C	D	I	E	F	G	H	J	K	L	M	
88	15	5	5	8	3	6	0	9	0	0	6	1	A
19	50	5	5	4	12	10	5	15	0	0	3	2	B
9	13	25	2	2	3	2	0	1	0	0	3	0	C
4	3	2	49	22	0	0	0	0	0	3	2	0	D
7	1	2	22	92	0	1	0	1	0	0	3	0	I
6	16	3	2	2	6	5	0	2	0	0	0	0	E
8	10	5	1	0	12	8	1	0	0	1	1	0	F
1	8	0	0	0	0	0	18	10	0	0	0	0	G
16	10	1	0	0	4	1	1	76	0	0	1	2	H
0	1	0	0	0	0	0	0	0	20	0	0	0	J
0	0	0	7	0	0	0	0	0	0	49	0	0	K
4	5	3	5	4	0	4	0	0	0	0	3	0	L
2	9	0	1	0	0	0	0	3	0	0	0	5	M

roundabouts. This reduces the number of classes to 9, while the classification rate increases by 5%.

The negative point of this classification is now *B: traffic* class with a classification rate of around 20%. This is due to the broad range of driving situations included in this class. Expert drivers emphasize that traffic situations varied in behavior according to road type. Consequently, we create clusters depending on road type, and *B: traffic* class instances are separated into 3 classes, *i.e.*, *E: urban speed reduction*, *H: speed limit* and *B: traffic*, respectively, as can be seen in table 3.8. This solution ensures a final global classification rate of 74%.

Table 3.8: Traffic class clustering

Clusters	New class
Road type = urban	<i>E: urban speed reduction</i>
Road type = main road	<i>H: speed limit</i>
Road type = highway	<i>B: traffic</i>

Finally with all the modifications of labeling that we made, the recognition accuracy increased from approximately 50% to 74% as shown in table 3.9. LLR even achieved a slightly better result with 76% recognition rate.

Table 3.9: Classification rate with final database

Classification rate (%)														
Class	All	A	B	C	D	E	F	G	H	I	J	K	L	M
TP rate	74	57	58			75		51	68	87	95	96		55
Instances	913	169	19			236		37	136	219	21	56		20

This final result of 74% is obtained with a 10-folds cross validation on our database of 913 events of the **Database 1**, with a false positive rate of 6.7%.

To check the generalization skill of the proposed approach, further collect a novel dataset of driving data using the same vehicle. This additional dataset consists of 242 events from **Database 2**. The 913 events of the initial database are used as a training set for the algorithms and the 242 events as a test set. In these conditions and with the same selected attributes, C4.5 achieves a 70% recognition rate, which is quite close to the 74% achieved on the initial database. LLR achieves a recognition rate of 71%. If we use 10-folds cross validation with the **Databases 1 and 2**, we have the same order of magnitude of the classification rate.

This recognition rate is rather good given the number of classes and the fact that the used only in-vehicle data. Moreover, it is quite understandable that driver behaviors faced with different events, *e.g.*, braking for a roundabout or truck size constraint event, are similar. This was expected from our expert drivers' experience, and is why the classification rate, validated with a second database of 242 events, is acceptable for our goal. Finally, this work showed that basic features, especially maximum and minimum speed, are the most important for classification of *Throttle=0* events. This initial work on *Throttle=0* events has allowed us to build and validate our approach, and thus proceed to *Throttle>0* events classification.

3.4.2 Impact of vehicle and driver on Throttle=0 events recognition

Database 2 enables us to evaluate the impact of driver and vehicle on driving events recognition. The database of 11 drivers on the same distribution cycle with the same FL vehicle contains 1689 *Throttle=0* events. The classification is 67% with 10-folds cross validation. In addition we can also estimate vehicle impact by learning with the initial database (the same 913 events from **Database 2**), and by testing with the 272 events acquired with the same reference driver, with the long-haul vehicle, on the two selected cycles. The classification rate is 72% with the most discriminant features and the two algorithms. This is logical since the parameters that describe the events most effectively are maximum speed, minimum speed and delta speed, and these features do not change depending on the vehicle (along with the road type and stop features). This new database allows us to validate classification work of *Throttle=0* events, and to observe that the vehicle has no major impact on *Throttle=0* events classification.

3.4.3 Throttle>0 events classification

Database 2 includes 2442 *Throttle>0* events distributed over 6 classes as explained in 3.1.2: *A: maintain a speed limit, B: reach a higher speed limit, C: urban take-off, D: main-road or highway take-off, E: adjust for stop and F: adjust for other reasons*. Unlike *Throttle=0* events, we rapidly achieve quite satisfying results thanks to the inter-class dissimilarities and intra-class similarities that these classes present. The principal task is therefore to estimate the best possible compromise between the features and the classification rate. Table 3.10 summarizes the classification results

with the two selected subsets of features: one subset with eight features gives the best classification rate, while the other gives a slightly lower classification rate but with only four features. As before, we used the two learning algorithms, namely C4.5 and LLR. With respect to $Throttle=0$ events study, both the two algorithms provided similar classification accuracies.

Table 3.10: Classification results depending on features

Features	C4.5 algorithm	Linear logistic regression algorithm
	Classification rate (%)	
Δ speed, standard deviation of speed, minimum speed, road type.	92.9	93.2
Δ speed, standard deviation of speed, minimum speed, road type, maximum acceleration, fuel consumption, mean percentage torque, duration of sudden speed change.	93.2	94.2

Table 3.10 presents the final classification rate, which is around 93% for $Throttle>0$ events, with a false positive rate of 4.3%. This performance is much higher than that achieved for $Throttle=0$ events, as the selected features prove to be discriminating for the 6 $Throttle>0$ classes. For example, minimum speed and delta-speed are quite different for a *C: urban take-off* in comparison with an *A: driving at speed limit* instance. This implies less confusion during the classification process.

3.4.4 Summary of events recognition from real driving data

We believe that the recognition rates achieved for $Throttle=0$ and $Throttle>0$ events are sufficient for our aforementioned application for two reasons. First, given the raw nature of the data, our expert drivers recognize that some events can be confused, *e.g.* an exit ramp and a speed limit. Secondly, the confusion matrix indicates that instances misclassified are actually situations where driving could be similar. For example, the classification method never considers a toll to be a traffic event. Since the goal is to reproduce driving situations in simulation, identified confusions can be accepted since they merge similar driving approaches.

An interesting question is how achieved performance compares with state of the art. Unfortunately, a direct comparison of achieved performance with the state-of-the-art is not feasible since both the driving data and the driving events to be recognized, considered in this work, are different from those found in the literature. However, instead of the two learning algorithms chosen, we also test the instance-based learning k-NN method used in [6] and [7]. We conduct a series of experiments varying k from 1 to 10 in order to determine the best value of k . We found that, where $k = 4$, k-NN achieves the best classification rate, namely 73% for $Throttle=0$ events and 91% for $Throttle>0$ events, with a false positive rate of 6.2% and 4.5%,

respectively. These results prove to be close to the accuracies achieved by the two learning algorithms, and remain comparable to the performance observed in the literature, even though the compared works are not identical since both event type and chosen driving data are different. Thus, we can now recognize driving events thanks to real driving data with selected features for both types of event.

3.5 Driving events recognition in digital roads

Automatic recognition of driving events, as developed in the previous section, is intended to be implemented in our simulation tool. In this section, we introduce the notion of digital roads, and present the results of applying the previous learnt models to simulation data to check their relevancy and ability to recognize driving events.

3.5.1 Digital roads definition

Instead of real data, the simulation tool used during our truck design process takes as input what is called a *digital road*. These roads are actually speed *vs.* distance square signals, where each value is a speed target. At each distance, the slope, road type and stop information is also known. An example of a short digital road is shown in Fig. 3.9. As can be observed, actuator information (throttle, brake, retarders, and coasting) is lost with this kind of data. As a result, we decide that each speed target change is an event, since we have no information on use of actuators on a digital road. Classification is a two-step process: identify the type of event, and recognize the event.

To identify the type of event, *i.e.*, estimate whether it is a $Throttle=0$ or $Throttle>0$ event, two rules are created. First, if the final speed is in the range $[Speed_{initial} - 10km/h; Speed_{initial} + 10km/h]$, the event type depends on the slope. For a positive slope, the event will be of type $Throttle>0$, while for a negative slope the event will be of type $Throttle=0$. For example, if the vehicle is on a road with a significant negative slope and the vehicle does not accelerate greatly, the driver will intentionally slow the vehicle down using brakes or retarders: this is then a $Throttle=0$ event. Secondly, if the final speed is outside this range, either the final speed is superior to the initial speed, in which case it is a type $Throttle>0$ event, or conversely it is a type $Throttle=0$ event.

To identify the event, we use the adapted classification method. In other word we apply the decision tree built with the real database for each type of event ($Throttle=0$ or $Throttle>0$). However, the input features are those that we can compute on the simulation data, namely maximum and minimum speed, $\Delta speed$, road type, and stop.

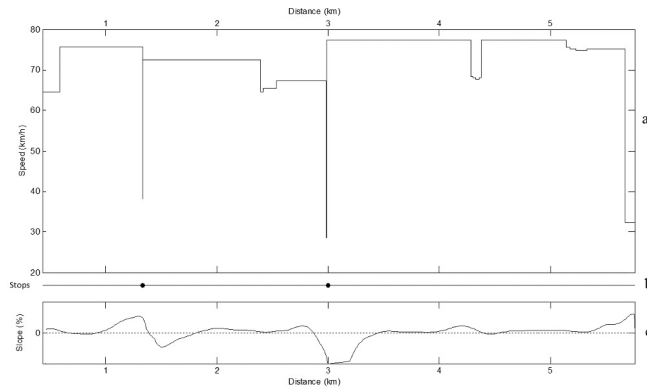


Figure 3.9: Example of a digital road: *a* is the square signal of speed vs. distance. On graph *b* each point represents a stop. *c* is the slope vs. distance function.

3.5.2 Database

We construct the following three databases to test the classification method on different kind of data:

- Urban database: 11 digital roads of 5km of urban road. The roads mainly include take-off, higher speed limit, maintain speed limit and roundabout events.
- Main road database: 11 digital roads of 6km of a main road. The roads mainly include take-off, higher speed limit, maintain speed limit and roundabout events.
- Highway database: 11 digital roads of 48km of a highway road. The roads mainly include take-off, higher speed limit, tolls, exit ramps, stop and maintain speed limit events.

3.5.3 Results

Since training is conducted with the real data, the simulation data is actually a test set for the decision trees for $Throttle > 0$ and $Throttle = 0$ events. We conclude above that the most discriminating features are basically the most common ones, including speed and road type features. Consequently, the classification rate is expected to be close to the rates previously reached. Tables 3.11, 3.12 and 3.13 present the confusion matrices for classification on the three databases. The different classes are represented by letters to ensure readability. Fig. 3.10 depicts an example of driving events identified for a main road section.

Classification rate for urban, main road and highway databases are around 80%, 81% and 90%, respectively. Similar to the work with real data, the confusion matrices disclose the errors between situations that could be considered as similar by drivers in terms of behavior.

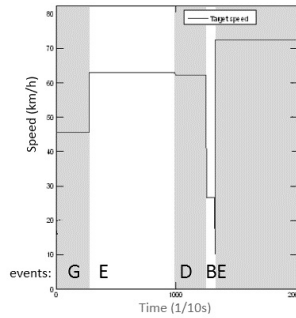


Figure 3.10: Example of identified events on a digital road. Letters represent the different events.

Table 3.11: Confusion matrix for urban section

A	B	C	D	E	F	G	H	classified as
139	0	14	0	23	0	0	0	A
2	49	0	0	0	0	0	0	B
8	17	18	0	0	0	0	0	C
2	0	0	8	0	0	0	0	D
0	0	0	0	70	0	0	0	E
0	0	0	0	0	3	0	0	F
0	0	0	0	0	8	0	0	G
0	0	0	0	0	0	0	11	H

Table 3.12: Confusion matrix for main road section

I	J	A	B	C	D	E	G	classified as
1	0	0	0	0	0	0	0	I
0	1	0	0	0	0	0	0	J
0	0	132	2	16	0	24	0	A
0	0	1	29	0	0	0	0	B
0	0	0	5	0	0	0	0	C
0	0	0	0	0	11	0	0	D
0	0	0	0	0	0	24	0	E
0	0	0	0	0	0	0	11	G

3.6 Conclusion

The twofold objective of the classification of driving events was to construct a learning-based classification that would work both with real and simulated data. Classification of $Throttle=0$ driving events is no easy task due to the inter-classes similarities and intra-class dissimilarities revealed by the Fisher linear discriminant measurement. For this reason, the final classification rate of 74% is a satisfactory result for our aforementioned application, given the nature of the data for $Throttle=0$ events. $Throttle>0$ events are classified based on the same principle and the resulting classification rate, around 95%, is better. Another problem was to adapt

Table 3.13: Confusion matrix for highway section

D	K	L	H	G	E	M	J	A	classified as
2	0	0	0	0	0	7	0	0	D
1	21	0	0	0	0	0	0	0	K
0	0	20	0	0	0	0	0	0	L
0	0	2	11	0	0	0	0	0	H
0	0	0	0	32	0	0	0	0	G
0	0	0	0	1	0	0	0	0	E
0	0	0	0	0	0	9	0	0	M
0	1	0	0	0	0	0	8	9	J
0	11	0	0	0	36	0	2	561	A

the decision tree learnt with real driving data to simulated data. However, since the most common features, *e.g.*, speed, are also the most important features for classification, we achieve a recognition rate of between 80% and 90% for the simulated data. These results suggest that the model can be transferred from real data (events defined by the use of actuators) to simulated data (events defined by changes in target speeds). It is worth noting that in both applications, real or simulated, events are not defined using a fixed-length duration (time or distance) or lateral information. As the driving events are now recognized, our next step is to analyze the difference of driver behaviors facing the same driving event and to characterize the main features for rational driving.

Correlation of driver behavior parameters with RDI

We have seen in the previous chapter that recognition of driving events is possible with the two selected algorithms. As it was presented in Fig. 1.10, the second part of this work concerns driving behavior in terms of rational driving in the scope of the driving events. Concretely, we aim to find driving parameters that are correlated to rational driving. Our goal here is to study how different driving behaviors impact the fuel consumption and the average speed and thereby characterize rational truck driving through a set of practical driving features.

For this purpose, we first introduce an indicator, namely *rational driving index* (RDI), which enables us to quantify different driving styles in terms of rational driving. We then select a set of state-of-the-art driver-dependent features within the truck context and experiment them for their discriminating skills in terms of driving behavior, using real driving data collected from 34 drivers who piloted the same truck on an extra-urban road and on a highway road. We then study, through a simulation tool, the impact of selected parameters in terms of fuel consumption and average speed, all other parameters being equal.

Concerning the extra-urban road, we will focus on the two major driving parameters identified during the previous step, namely passing speed in roundabouts and average deceleration during the braking phase. Given the fact that France has half the total number of roundabouts worldwide [76], crossing a roundabout in an extra-urban environment is a driving situation particularly representative of merchandise transport companies which deliver regionally. Simulation is useful to evaluate the impact of one driving parameter at a time while fixing all the other ones. This is not possible when driving on a real open road where the environment keeps changing. Concerning the highway road, we will focus on the percentage of coasting distance since coasting is a key parameter of the definition of rational driving. Finally, we compare the RDI to a cost model for this specific vehicle, to prove that the RDI is relevant to evaluate rational driving capacity.

The contributions are fourfold:

- We define an indicator of *rational driving* instead of *economic driving*. This indicator, namely *RDI*, enables us to measure to what extent driver behavior is compliant with rational driving. Rational driving aims to decrease the fuel consumption while maximizing authorized commercial driving speed for merchandise transport tasks;

- We analyze a set of driver-dependent features which contribute characterizing driver behavior, in particular with respect to the rational driving;
- Using the major features identified as particularly sensitive to the driver behavior, we further study and quantify, through simulations, their impact on rational driving, *i.e.*, in terms of fuel consumption and average speed.
- We propose an evaluation of the RDI by comparing it to a cost model expressed in €.

The rest of the chapter is organized as follows. Section I defines our rational driving indicator. Section II presents the data collection and shows that there are major differences between drivers with respect to the fuel consumption and the average speed. Section III presents the driver classification using machine-learning algorithms. Section IV displays the measures of correlation between selected driving features and rational driving index on extra-urban road and on highway road. Section V and VI focus on the simulation-based impact study of different driving behaviors on a representative roundabout and on a toll, respectively. Section VII compares the RDI to a cost model. Finally, in section IV, we will conclude the chapter.

4.1 Rational driving index

First of all, we aim to create an indicator able to rank drivers depending on their ability to drive rationally. Maincent [3] defines the notion of *rational driving as the full exploitation of the potential of the vehicle to drive more fuel efficiently while respecting the delivery delays*. We go one step further in suggesting that one could achieve fuel saving while maximizing the authorized average speed. This implies in particular that drivers can reduce their fuel consumption without losing time traveling the same distance for their deliveries. Fuel consumption should not be the only parameter to be taken into account for rational driving. We propose a rational driving index to enable comparison between two driver styles in terms of rational driving, and we propose a coefficient used in [77] to compare two vehicles. This *Rational Driving Index* is referred as to *RDI* and is defined as the ratio of the average speed on the fuel consumption for a given journey:

$$RDI = \frac{\text{Average speed (km/h)}}{\text{Average fuel consumption (L/100km)}} \quad (4.1)$$

As it can be seen from the equation, the RDI increases when the fuel consumption is reduced and/or when the average speed increases. The higher the RDI on a fixed journey, the better the driver with regard to the rational driving. For a given journey and similar exterior conditions, this index thus enables us to sort drivers in terms of driving rationality, the best ones being those who reduce the fuel consumption of the vehicle without losing time, or even saving time.

Using the same weighting for average speed and average fuel consumption is coherent since average speed and fuel consumption have the same order of magnitude (especially true for the heavy trucks context, on the contrary to cars). Moreover, their equal weighting underlines that both terms have the same importance.

4.2 Driving data collection

While the rational driving index *RDI* enables us to compare different drivers in terms of the quality of the rational driving, we do not know the main driving parameters which contribute characterizing such rational driving. In this section, we aim to answer this question. For this purpose, we collected a set of real driving data from 34 different drivers in approximately the same driving conditions, *i.e.*, using the same vehicle on the same road with similar traffic conditions, and we studied the correlation of several state-of-the-art driving features with respect to *RDI*. This database is called **Database 3** and the details of how this data collection was carried out are given below.

4.2.1 Vehicle

Specifically, 34 drivers were asked to drive a Renault Trucks D-wide vehicle as illustrated in Fig. 4.1. It is a 280hp Euro6 Heavy Multi-Purpose vehicle, representative of vehicles for a regional distribution usage. The vehicle was equipped with a robotized gearbox, two sets of engine retarders (exhaust and compression) and loaded to 14.5 tons. The drivers were asked to drive normally but without using kick-down and cruise control, and always keeping the robotized gearbox in auto mode. These instructions were provided with the aim of isolating the driver features from those of the vehicle with respect to the impact on fuel consumption. Indeed, when using the robotized gearbox in auto mode while excluding the cruise control usage, all the drivers must control the vehicle speed only with the throttle pedal, brake pedal and retarders. In-vehicle driving data were recorded using three sensors, namely GPS data (latitude, longitude *etc.*), CAN data (vehicle speed, fuel consumption, brake pedal *etc.*) and video data synchronized to the CAN-data logger. The video was used to select similar runs and remove those presenting high density traffic for instance.

4.2.2 Drivers

We carried out his data collection with 34 different drivers in order to have more data and especially more reliable data than with the previous data collection. Indeed we carried out a data collection in 2012 with 11 drivers, and we observed that there were not enough data to draw pertinent conclusions on driving parameters regarding



Figure 4.1: The Renault Trucks D-wide vehicle, a heavy multi-purpose vehicle, used to acquire real driving data. This vehicle is quite representative of vehicles used for regional distribution.



Figure 4.2: Camera for video-recording used during the data collection.

rational driving. Moreover the drivers were internal workers and therefore they did not always have a driving behavior similar to external professional drivers.

Thus, the objectives of this data collection were:

- We involved more drivers in order to collect more data, with a wider population targeted not always familiar with our test environment (unlike previous data collections). It will allow us to identify and recognizing driving behaviors.
- During the data collection, there will be one reconnaissance lap (with an internal observer) and one real lap (driver alone in the truck) for each driver. Thus, we will have more data, but especially more reliable data because the driver is alone during the second lap (the “white coat effect” is removed) and moreover he/she already knows the road. This gives the driver the possibility of improving the fuel consumption of the vehicle if the driver is concerned with this, and thereby increase the disparities between drivers.
- Vehicle model is available in our internal simulation tool.

Among the 34 drivers there were 4 internal drivers, 10 temporary workers from our proving ground LVPG and 20 temporary workers from an external company. There was one woman among them, equivalent to 3% of the population. The age pyramid averages at around 56-65 years old, which is older than the average of all professional truck drivers in France (centered around 40-54 years old). Fortunately, as presented in Fig. 4.3 and Fig. 4.4, age and experience are not correlated to rational driving. Indeed, the *RDI*s calculated on the whole cycle are not correlated to driving experience or the age of the drivers.

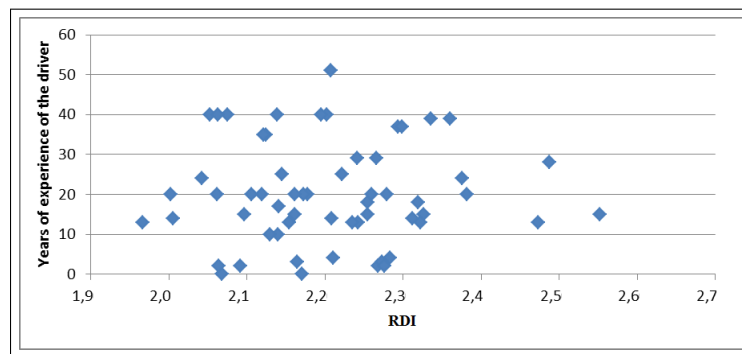


Figure 4.3: Years of driving experience of drivers *vs.* *RDI*.

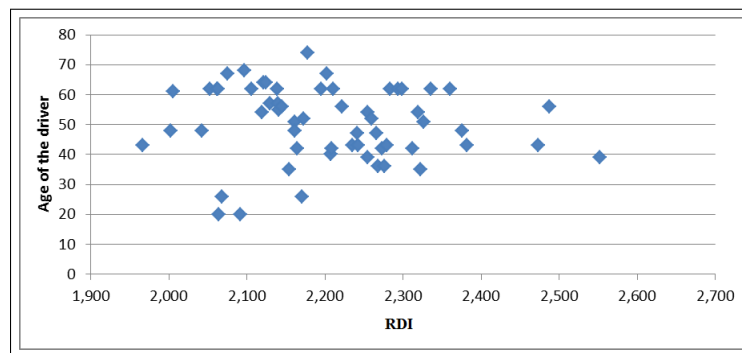


Figure 4.4: Age of drivers *vs.* *RDI*.

Most of the time in their work, the drivers drive every day, with tractors and rigid trucks, on regional or national roads. A majority of them drive less than 90000km per year. Once the drivers finished the second lap, we asked them some questions in order to know if they are sensitive to fuel consumption and to some other parameters. Finally on average they are all interested in all the parameters except finishing their work early, therefore this questionnaire is not a help to classify drivers, as we can see in Fig. 4.5. These 34 drivers thus cover an interesting part of

the truck driver population since they do not all have the same driving profile and work experience.

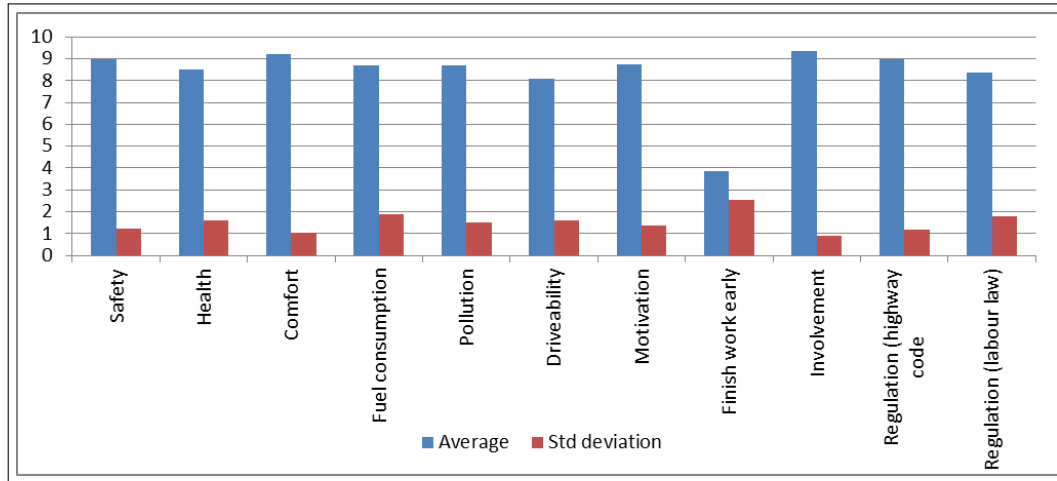


Figure 4.5: Interests of the drivers. They gave their answer on a scale from 1 to 10 for each subject (10=highest mark).

4.2.3 Road

We chose a representative road of regional distribution presented in Fig. 4.6, *i.e.* mainly urban or extra-urban road with a few stops. In this chapter we will focus on the main road section and the highway + toll section framed in Fig. 4.6. The other sections framed in green are not studied in this report, but are part of the road presenting interesting characteristics for the study of fuel consumption and average speed.

The 34 drivers drove this path twice, therefore there are 68 runs all in all, but only 58 without errors. These 58 runs can be used to compare drivers on the whole cycle because drivers follow exactly the same path. 67 runs are useful to compare driver behaviors on selected sections (not on the whole cycle), such as the main-road section presented above. And one run is not useable because of intense snowfall. The database represents 169 hours of driving and 8443 km.

4.2.4 Differences in fuel consumption and average speed between drivers

We know from previous studies such as in [3] that drivers have a major impact on the fuel consumption and average speed of the vehicle. This gap between drivers depends

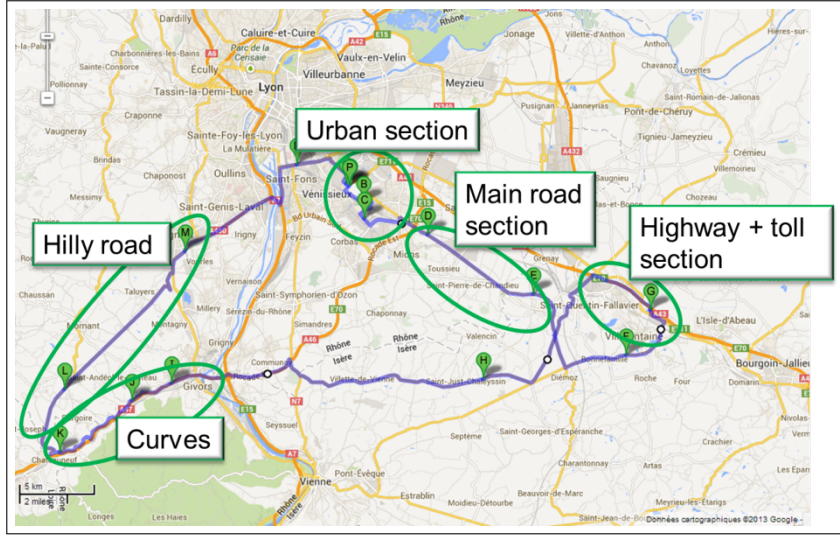


Figure 4.6: Road of the data collection. Framed roads are noticeable sections of the path.

on the truck and on the truck application. Indeed there will be fewer differences between drivers for a long haul usage on a flat highway, than on a distribution usage on hilly roads.

On the 67 comparable runs covering the whole cycle, we measure the gap between the average fuel consumption of each driver with the global average fuel consumption and equally for the average speed. The gaps on average speed between drivers are comprised between -7.2% and $+7.5\%$ of the average value. The gaps concerning fuel consumption are higher: between -10.8% and $+18.5\%$ of the average value. As expected, this data collection offers a wide disparity in fuel consumption results between drivers with an average speed quite similar between them.

If we remove the reconnaissance lap where the drivers did not know the road and where they were accompanied, the speed gap between drivers is narrower (between -7.1% and $+4.9\%$), whereas the gap in fuel consumption is similar (between -11.3% and $+18\%$) as presented in Fig. 4.7. The reconnaissance lap has hence achieved the desired effect since the width of the gap in fuel consumption is really high whereas the gap in speed is not very high, by comparison. Moreover, the drivers are alone in the truck during the real lap and they know the path, so it can be supposed that they drove more naturally.

Concerning rational driving analysis, we observe a distribution of the *RDI*s between 1.9 and 2.6. This distribution looks like a Gaussian one (RMSD=2.2 between the real distribution of *RDI*s and the theoretical Gaussian distribution obtained with the mean value of *RDI* μ_{RDI} and the standard deviation σ_{RDI}), with many drivers having an *RDI* close to the average value of *RDI*. We can separate intuitively

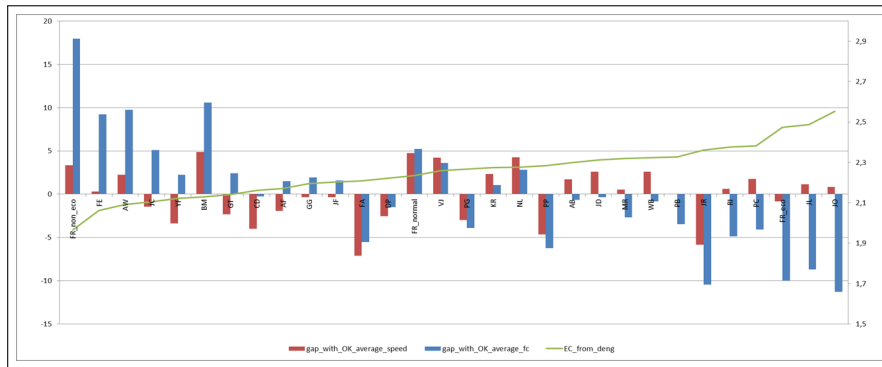


Figure 4.7: Gaps with average fc (blue) and average speed (red) for each driver (real lap only). RDI is represented in green. The best drivers are on the right.

the drivers into three classes with the thresholds $\mu_{RDI} - \sigma_{RDI}$ and $\mu_{RDI} + \sigma_{RDI}$. We compare the distribution of drivers' RDI obtained here with the distribution of similar vehicles from customer data. This data is extracted with an internal tool: Logged Vehicle Data (LVD). The distribution is based on 939 Premium Distribution data (*i.e.* 1373-20% of extreme values) and 2496 Midlum data (*i.e.* 3123 -20% of extreme values) and is presented in Fig. 4.8. As expected, the width of the distribution of the 34 drivers from our data collection is lower than the other distributions (Premium or Midlum). This is because we only have one trip with a unique usage, whereas the data from LVD come from many customers with different trips and different uses and also of course from more drivers. However, the distribution of the 34 drivers is centered on the same value as the other distributions, it means that the selected path is coherent with the truck application and the values of RDI obtained are relevant.

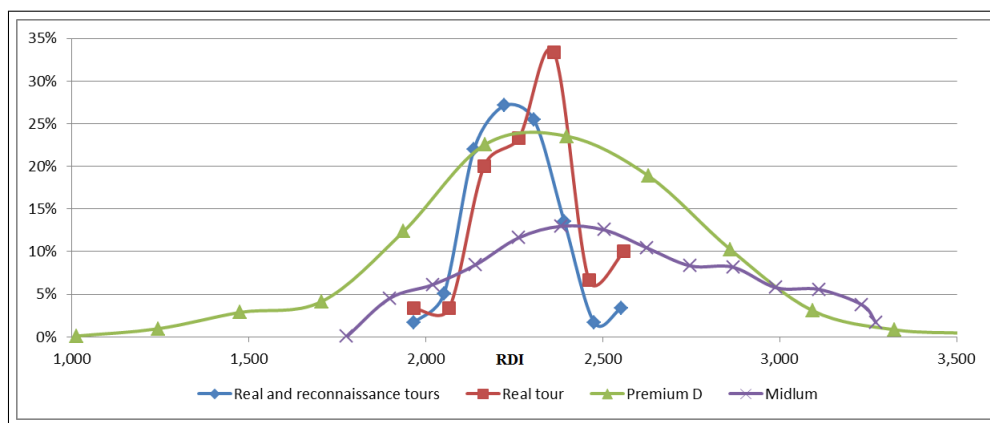


Figure 4.8: Distribution of RDI for the 34 drivers of the data collection compared to the distribution of RDI from customer data: 939 Midlum and 2496 Premium D.

Consequently, we know that our data collection is relevant and that we observe a wide dispersion between different drivers with the same vehicle on the same path. But another objective of the data collection is to estimate the dispersion of a unique driver on the same path with the same vehicle. An internal driver drove 8 times on the same road (the 16 first kilometers of the cycle, back and forth). He was asked to drive normally. Two runs are not relevant because of a big traffic jam and a windy day, respectively. Therefore we need to remove outliers and choose to study the dispersion of the driver on the 4 medium runs only, that is to say that we remove the two non-comparable runs (with the worse *RDI*s) and the two best *RDI*s runs too.

On these 4 runs, the dispersion is around 3% in fuel consumption, around 5% in speed and around 2,6% in *RDI*. These figures are higher than expected, however the standard deviation of the *RDI* for the 4 runs of the driver is really lower than for all the other drivers. On the first 16km one-way we are able to measure the *RDI* for each of the 33 other drivers, and to compare them to the *RDI* for the 4 runs of the same driver. The standard deviation of the *RDI* for the 33 drivers is almost 5 times that of the standard deviation of the *RDI* for the sole driver. Therefore, the differences obtained in fuel consumption between the drivers are indeed due to the differences in driver behavior.

Table 4.1: Average and standard deviation of *RDI* for the 4 times with the same driver and the 33 other drivers. It is measured on the first 16km of the cycle, one-way only.

Driver	Average <i>RDI</i>	Standard dev. <i>RDI</i>
Same driver (4 times)	1,643	0,031
33 other drivers	1,515	0,146

4.3 Hierarchical classification

We remind you that our methodology, as illustrated in Chapter 1 Fig. 1.10, is to recognize driving behaviors during the approach of the recognized driving events. We use the **Database 2** that contains driving events from 11 drivers on the same road with the same vehicle. We initially use two methods for driver classification: the first one is the classification of the drivers on all events, the second one is a hierarchical classification: we classify events first, then we separate instances depending on the result, and then we classify drivers for each event-database file. These two methods are illustrated in Fig. 4.9 and Fig. 4.10.

With the first method, the classification of the 2242 *throttle* > 0 event instances and 1689 *throttle* = 0 event instances gives poor results, *e.g.*, around 12% and 30%, respectively. Driver classification is not better with the second method.

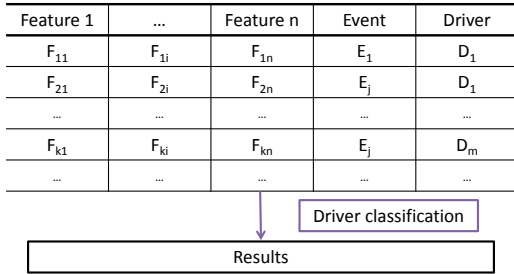


Figure 4.9: First method for driver classification.

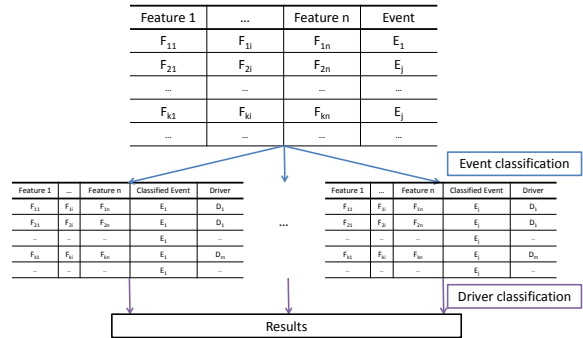


Figure 4.10: Hierarchical method for driver classification.

We then rank the drivers by their RDI, and separate them into three classes of equal frequency (4, 4, and 3 drivers in each group from the least efficient to the most efficient, respectively). The classification on these 3 classes of driver type improves the results up to an average rate of 40%, as illustrated in Fig. 4.11. However, the confusion matrix proves that the classification is still not relevant.

This classification is not conclusive because drivers do not behave homogeneously throughout the cycle, and the approach between different events of the same type but at different places cannot always be comparable. Given these results, we choose to select short sections of a few kilometers, and to study the driving behaviors on these sections instead of on the full cycle. The risk of driving behavior change for each driver is reduced with this approach, as well as the diversity of driving approaches for the same type of event.

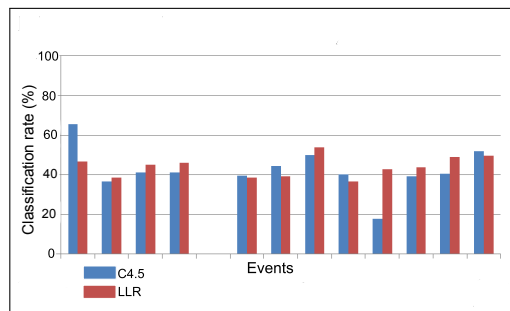


Figure 4.11: Classification rate of the driver type classification with C4.5 and LLR algorithms

4.4 Rational driving-related driving features

We focus our attention on two road sections: one extra-urban road and one highway road with a toll. Extra-urban road is chosen since drivers have a lot of impact on the fuel consumption in this environment. Concerning highways, we focus on toll events which are a key event because of the intrinsic high speed variation, and also because the overall events density is rather low. For each section, we select a part of the total cycle and measure the *RDI* and driving features for each driver.

4.4.1 On extra-urban road

The main-road section is a 6km road where the drivers' behavior on roundabouts can be studied easily. The selected section is presented in Fig. 4.12. Given that there are 3 roundabouts on a straight line road with low traffic, the main driving events on this road are the roundabouts.

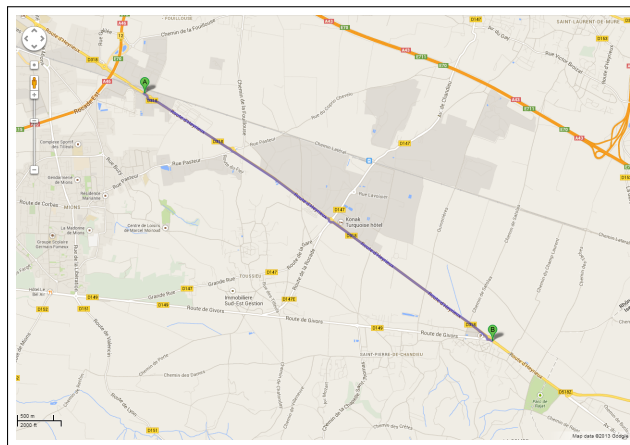


Figure 4.12: The extra-urban section which includes 3 roundabouts.

We select the driving data of 75 different runs by the 34 different drivers on this extra-urban road. The exterior conditions were approximately the same, and the drivers are sorted according to their *RDI* which gives hints on their ability to drive rationally. As it was expected, the distribution of *RDI*s presented in Fig. 4.13 looked like a Gaussian distribution (RMSD with a Gaussian distribution=7.8). Few drivers have a very low or very high *RDI* while most of them have a *RDI* close to the average *RDI* value. This *RDI* distribution enables us to separate intuitively the drivers into three different classes according to their driving rationality, namely *efficient*, *medium* and *non-efficient* drivers.

The next step is to find features that could contribute to a rational driving. Our study on previous works highlighted several state-of-the-art features with respect

to the rational driving. For a list of features correlated to the RDI , several state-of-the-art driving features are listed in the first column of Table 4.2. They are adapted to the truck context, *e.g.*, the threshold of acceleration for even driving is modified to an appropriate value. Their Pearson's correlation coefficients with its corresponding RDI are displayed in the second column. The Pearson's coefficient measures the linear dependence between two variables, giving a value between $+1$ and -1 inclusive, where $+1$ is total positive correlation, 0 is no correlation, and -1 is total negative correlation. The coefficient is defined by:

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X\sigma_Y} = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X\sigma_Y} \quad (4.2)$$

where $\rho_{X,Y}$ is the Pearson's coefficient between X and Y , $cov(X,Y)$ is the covariance between X and Y , σ_X is the standard deviation of X , μ_X is the mean of X and E is the expectation.

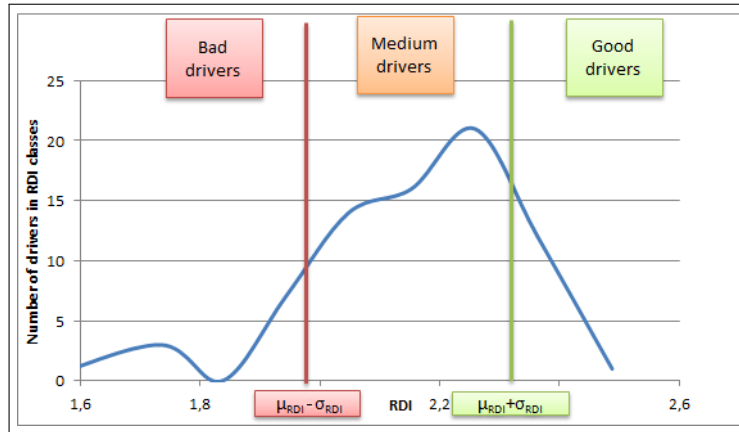


Figure 4.13: Distribution of the RDI s of the drivers on the 75 runs. We separated drivers in the three classes with the help of the range defined as: $[(\mu_{RDI} - \sigma_{RDI}); (\mu_{RDI} + \sigma_{RDI})]$ where μ_{RDI} is the average value and σ_{RDI} is the standard deviation of the RDI s.

As it can be seen from Table 4.2, although the correlation coefficients are not equal to zero, none of them displays a high Pearson's coefficient. It is thus not that obvious to conclude on their strong correlations. This result was expected since the driving data were collected on an open road. The environment was thus similar but not always equal, especially concerning weather or traffic. Wind and rain for example have an impact on the fuel consumption because they increase the drag force on vehicle motion.

It must be decided whether the correlation is significantly different from zero, *i.e.*, whether or not a non-zero correlation exists. The assumptions are that the

Table 4.2: Correlation study between driving features and *RDI* on extra-urban road

Driving features	Pearson's coefficient between the driving feature and <i>RDI</i>	p-value
Percentage of coasting time (%) [3]	0.13	0.260
Number of braking events [34]	-0.42	0.0002
Number of retarders events	-0.05	0.696
Percentage of time where the position of the throttle pedal is 100% (%) [34]	-0.10	0.416
Percentage of time evenly driving (%) <i>i.e.</i> $ Acceleration < 0.2m/s^2$ [34]	0.31	0.007
Percentage of time with extreme acceleration (%) <i>i.e.</i> $ Acceleration > 0.6m/s^2$ [34]	-0.32	0.005
Average speed in roundabout (km/h) [3]	0.38	0.0007

random samples, X and Y , are drawn from a population that follows a bivariate normal distribution, and the population correlation coefficient is zero: $\rho = 0$. If these assumptions are satisfied, the statistic

$$T = \frac{\rho_{X,Y} \sqrt{N-2}}{\sqrt{1-\rho_{X,Y}^2}}$$

follows a t-distribution with $N - 2$ degrees of freedom, where N is the sample size. The null and alternative hypotheses are:

$$H_0 : \rho = 0 \text{ (no correlation)}$$

$$H_1 : \rho \neq 0 \text{ (real correlation)}$$

We chose a α -level, *e.g.*, $\alpha = 0.05$ for a 95% confidence interval. The p-value is the probability of obtaining a test statistic result at least as extreme as the one that was actually observed, assuming that the null hypothesis is true. If $p < \alpha$, the correlation is significantly different from zero. We found that for the *number of braking events*, *even driving*, *extreme acceleration driving*, and *average speed in roundabouts*, the correlations are significantly different from zero with 95% confidence.

Maincent [3] also has low correlation coefficients between fuel consumption and the driving features that she tested. However, she choose to study qualitatively only 4 drivers that she selected, and to compare their behavior on very specific events. We propose a different approach based on the average values of the driving features for the three classes of drivers. Table 4.2 suggests five driving features with a higher Pearson's coefficient than we list in the first column of Table 4.3. Table 4.3 displays the difference between the average values of the efficient driver class and the non-

efficient driver class with respect to the average value of the medium driver class for each of these 5 driving features.

As it can be seen from this table, efficient drivers use on average more coasting, *i.e.* they tend to have more time where they use the kinetic energy of the vehicle only (they use neither the throttle pedal, nor the brake, nor the retarders). While driving more evenly with less high acceleration periods, efficient drivers also tend to brake less than the medium and non-efficient drivers. Finally, they have a higher crossing speed on roundabouts. This fact suggests that they anticipate better than other drivers, and thereby lose less velocity, *i.e.*, the kinetic energy of the vehicle. As a result, they inject less fuel in the accelerating phase which follows the roundabout. All these discoveries are coherent with the intuitive rules of rational driving and are thus promising. We will further quantify the impact of two driving features, namely the coasting time and the crossing speed, using simulations.

Table 4.3: Difference in % between driver types for selected driving features. Reference is medium *RDI* drivers.

Feature	Driver type		
	Lowest <i>RDI</i> drivers	Medium <i>RDI</i> drivers	Highest <i>RDI</i> drivers
Percentage of coasting time	-22.0	0	+14.3
Number of braking events	+73.8	0	-13.3
Even driving	-4.4	0	+6.5
Sudden speed change driving	+8.4	0	-73.4
Average speed in roundabouts	-13.7	0	+20.8

4.4.2 On highway

We also select a 4.5km highway road which includes an entering lane with a curve, 3km in a flat straight line and a toll gate where the drivers must stop. On this section, the distribution of the RDIs of drivers is equally Gaussian (RMSD=5.98, normalized RMSD=9.58%) as illustrated in Fig. 4.14. The majority of drivers have a medium behavior with respect to rational driving.

Similarly as in the study of the extra-urban section, we measure the correlation between the driving features and RDI. The results are presented in Table 4.4. The correlation is still low for the same reasons as before, especially for the number of retarders events, and the percentage of time with extreme acceleration. This last feature is indeed less important on a highway road since the acceleration phases

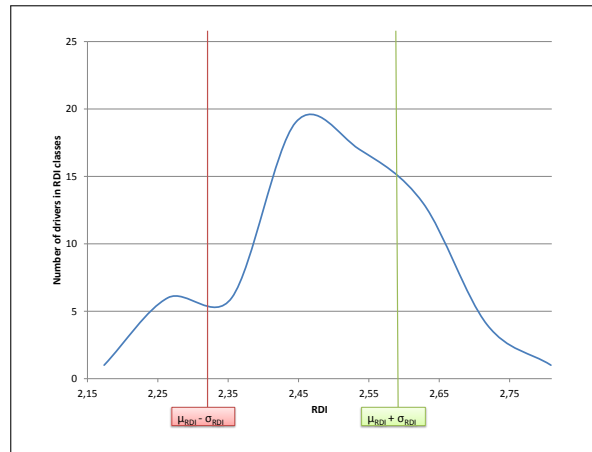


Figure 4.14: Distribution of RDIs on Highway section.

are less numerous, and are limited to the entrance of the highway. The p-values show that for the *percentage of coasting time*, *number of braking events*, *evenly driving*, and *percentage of time where the position of the throttle pedal is 100%*, the correlations are significantly different from zero.

Table 4.4: Correlation study between driving features and RDI on highway

Driving features	Pearson's coefficient between the driving feature and RDI	p-value
Percentage of coasting time (%) [3]	+0.32	0.007
Number of braking events [34]	-0.36	0.003
Number of retarders events	-0.01	0.939
Percentage of time where the position of the throttle pedal is 100% (%) [34]	-0.35	0.003
Percentage of time evenly driving (%) <i>i.e.</i> $ Acceleration < 0.2m/s^2$ [34]	+0.31	0.011
Percentage of time with extreme acceleration (%) <i>i.e.</i> $ Acceleration > 0.6m/s^2$ [34]	-0.14	0.247

Focusing more specifically on the toll approach, which is a key event on highway, behavioral differences are principally translated in terms of anticipation, braking, and utilization of the kinetic energy of the vehicle. The corresponding features, as in Maincent [3], are the distance of the event (*i.e.*, anticipation), the distance of coasting, and the continuity of braking. Table 4.5 presents the percentage of the driver population approaching the toll with only one braking event, and those who need several braking events and accelerating phases to reach the toll at the appropriate speed. Drivers who brake and accelerate many times are representative

of a lack of anticipation. We see in this Table that efficient drivers tend to have one event unlike medium and non-efficient drivers. However, most non-efficient drivers also approach the toll gate with one event so it is difficult to find a coherent conclusion on this driving feature.

Table 4.5: Percentage of the driver population having a continuous event, or not, during approaching the toll gate.

	Non-efficient drivers	Medium drivers	Efficient drivers
Continuity in braking, <i>i.e.</i> , 1 unique braking event for the toll	55.6%	39.5%	73.3%
Non continuity in braking, <i>i.e.</i> , 2 or more braking events for the toll	44.4%	60.5%	26.7%

Table 4.6 displays the differences between the average values of the efficient driver class and the non-efficient driver class with respect to the average value of the medium driver class for each of the 5 driving features with the highest Pearson's coefficient. We also add the percentage of coasting distance during the approach of the toll since this feature is the one representative of coasting utilization that we can implement in the simulation tool. As it can be seen from Table 4.6, efficient drivers use on average more coasting, *i.e.*, they tend to have more time where they only use the kinetic energy of the vehicle. While driving more evenly, efficient drivers also tend to brake on average less than the medium and non-efficient rational drivers. However, the acceleration features are not relevant due to the lack of acceleration phases on this road. Indeed, as it was expected given the low Pearson's coefficient, the percentage of time with extreme acceleration and the percentage of time where the position of the throttle pedal is 100% are, on average, not differentiating on this highway road. Finally, efficient drivers tend to approach the toll gate with only one braking event. This fact suggests that they anticipate better than other drivers. As a result, they do not have to re-accelerate before toll and hence they do not need to re-inject fuel unlike other drivers.

Since the conclusions are less obvious than on extra-urban road, we will focus on the attribute revealing the maximum gap between efficient and non-efficient rational drivers, *i.e.*, coasting. In the next step of our work we will quantify the impact of this driving feature with respect to rational driving with the help of our simulation tool, all other parameters being equal.

Table 4.6: Difference in % between driver types for selected driving features on highway road. Reference is medium *RDI* drivers.

Feature	Driver type		
	Lowest <i>RDI</i> drivers	Medium <i>RDI</i> drivers	Highest <i>RDI</i> drivers
Percentage of coasting time	-8.96	0	+60.84
Percentage of coasting distance during take-off approach	-48.34	0	+38.91
Number of braking events	+24.59	0	-28.43
Percentage of time where the position of the throttle pedal is 100%	-6.08	0	-36.78
Evenly driving	-0.60	0	+3.96
Sudden speed change driving	-15.10	0	-25.50

4.5 Simulation-based impact study of driving features when crossing a roundabout

The last section shows that the five driving features help to discriminate drivers in terms of rational driving on the extra-urban road. In this section, we use simulation to provide insights into the impact of these features on rational driving. Indeed, the simulation gives us the possibility of quantifying the impact of one driving feature on the rational driving, while fixing all other variables. This is not possible in real driving experiments in open environments. Specifically, we choose to evaluate the impact of two of the previous five driving features, namely the coasting time and the crossing speed on roundabouts. While the coasting time is related to one of the main rules of rational driving, the crossing speed in roundabouts impacts the average speed for transportation companies. It must be noted that roundabouts are a very common road infrastructure in Europe that drivers face frequently.

4.5.1 Selection of a representative roundabout and its representation on the simulation tool

The input of the simulation tool is a road which depicts a roundabout. Therefore we have to find a real roundabout representative of extra-urban driving and to create its digital equivalent. The road where we carried out the data collection includes several roundabouts. We select one of them according to several criteria, *i.e.*, low traffic, high visibility, same road type before and after the roundabout (*i.e.*, extra-urban road), and straight direction, such that drivers only need to concentrate on their driving and do not need to pay attention to road signs, traffic or turning directions. The third roundabout is the best compromise with respect to these criteria. Fig.

4.15 gives a Google Earth picture [78] of the selected roundabout and shows the driving direction of the selected infrastructure.



Figure 4.15: View of the selected roundabout and the driving direction.

We have to characterize the selected roundabout in order to build its digital representation using several parameters, *e.g.*, the diameter of the roundabout, and the distance with the previous and the next roundabout. The diameter of the selected roundabout is 47 meters which is an average value, because some drivers cut the corner on the turn whereas others drove on the outside of the turn. This average diameter enable us to compute the maximum speed to get around this roundabout. The maximum lateral acceleration tolerated for this Heavy Multi-Purpose vehicle is

$$A_{max} = 0.4 * g \quad (4.3)$$

where g is the Earth's standard surface gravity in m/s^2 . This value is the limit before rollover for that specific truck. In this case, the maximum speed to get around this roundabout is

$$S_{max} = \sqrt{A_{max} * R} \quad (4.4)$$

where R is the radius of the roundabout. This gives:

$$S_{max} = 34.6 \text{ km/h} \quad (4.5)$$

The distances with the previous and the next roundabouts are 1810 and 2940 meters, respectively. We impose the initial and final speed at 0km/h in order to concentrate our study only on the selected roundabout. Once given all these characteristics, we are able to build the digital representation of this roundabout as illustrated in Fig. 4.16. This will be the input of the simulation tool.

4.5.2 Simulating driver behaviors on the digital roundabout

The simulation tool includes three components, namely the vehicle, the road, and the driver, all of them having a significant impact on the fuel consumption and the average speed as illustrated in Fig. 4.17. For a given vehicle, the driving features

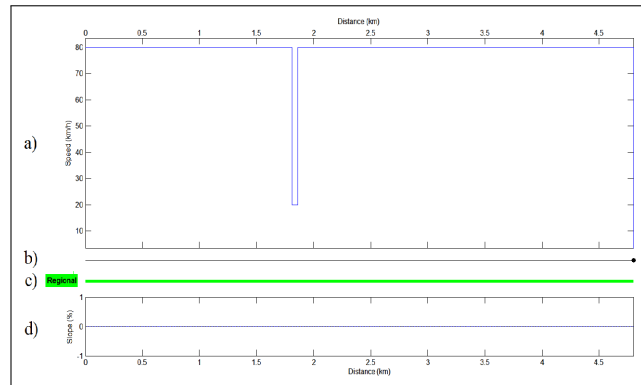


Figure 4.16: Selected roundabout model for simulation. a) is the speed target vs. distance. On b), the black point represents the final stop. c) is the road-type. And d) is the slope vs. distance.

whose impact we aim to evaluate on rational driving are the crossing speed and the deceleration, respectively. We try different target deceleration values on the driver side, and various roundabout crossing speeds on the road side.

The different parameters of various trial-runs on the simulation tool are:

- The cruising speed: 60km/h, 70km/h or 80 km/h. It describes the cruising speed that the driver wants to reach and to keep between the roundabouts. On this road the legal speed limit is 80km/h. However, we observed that the drivers rather targeted a cruising speed between 60 and 80 km/h. We test these three cruising speeds to keep to with real driving behaviors and to evaluate the impact of the selected driving features with respect to rational driving given an input cruising speed;
- The passing speed: 0, 15, 17, 18, 20, 22, 24, 34.6 km/h. 0km/h is the extreme speed value illustrating the case where the driver must stop at the roundabout because the driver did not anticipate properly for example. 34.6 km/h is the theoretical maximum speed that the vehicle can reach when turning on the roundabout as we calculated previously. The other speeds are representative of the real measured data;
- The configuration in which the roundabout is not simulated. In this case, drivers do not need to brake. They only have to keep the cruising speed for the whole distance. The results achieve in this configuration can be used as reference values, *e.g.*, to quantify the impact of the roundabout itself;
- The deceleration by braking: -0.6, -0.8, -1.2, -2 m/s². These values are representative of real measured data. These values enable us to interpret driving styles: the lower the deceleration, the more the driver uses coasting.

All combinations of the previous four variables are simulated. Table 4.7 illustrates some of these combinations. They are representative of different driving

behaviors in terms of cruising speed, crossing speed and deceleration. Run 1 corresponds to a typical behavior with the mean crossing speed and the mean deceleration. This run is considered as the reference behavior. All the other runs, *i.e.*, 2 through 5, deviate from this reference run.

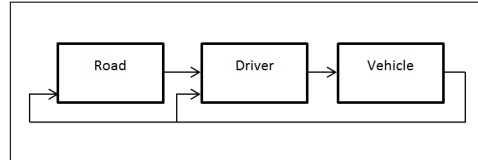


Figure 4.17: The main simulation parts in the simulation tool. Arrows represent exchange of data between driver, vehicle and road.

Table 4.7: Examples of different parameter combinations used in the simulations

Run number	Cruising speed	Roundabout	Crossing speed	Deceleration
1	80	Yes	20 km/h	-0.8 m/s ²
2	80	No		-1.2 m/s ²
3	70	Yes	34.6 km/h	-0.6 m/s ²
4	70	Yes	0 km/h	-2 m/s ²
5	60	Yes	15 km/h	-0.8 m/s ²

4.5.3 Impact of the simulated driving features on RDI

For each run on the digital representation of the 4.8km road section, we measure the simulated fuel consumption and average speed in order to compute the corresponding *RDI*. Fig. 4.18 illustrates several results from these simulations. As for real runs, simulated driver behaviors are sorted according to their *RDI*. Table 4.8 gives average speed, fuel consumption and *RDI* results for the five examples runs. Run 1 is considered as the reference. For each run, the gaps with the reference average speed and the reference fuel consumption are indicated in the second and third column of the Table, respectively. The corresponding *RDI* is in the last column.

4.5.3.1 Impact of the average deceleration

For a fixed couple of crossing speed and cruising speed, the only varying parameter is the average deceleration. Table 4.9 gives the Pearson's correlation coefficients

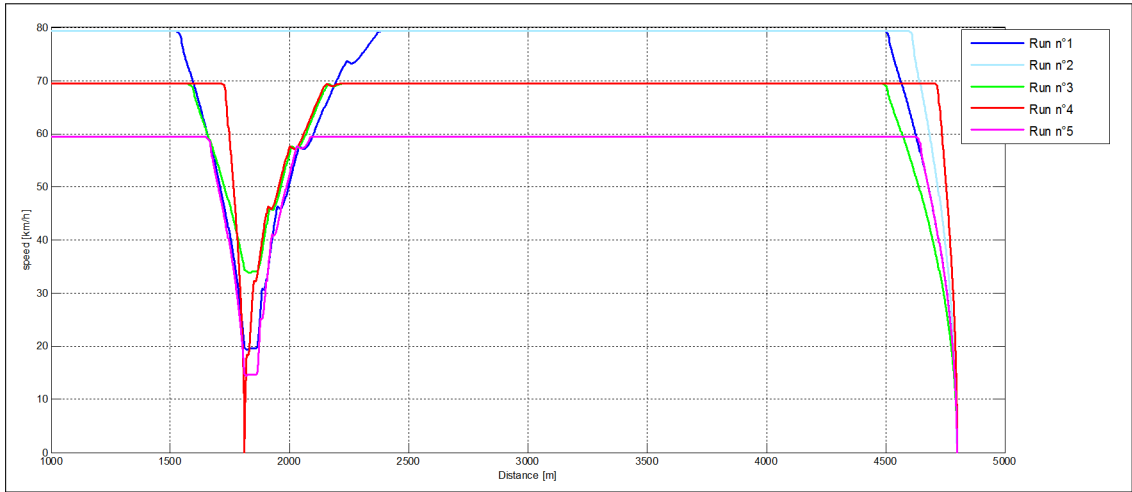


Figure 4.18: Speed vs. distance signals from the simulations for the five selected runs. We can observe the different cruising speeds, crossing speeds (at 1810m) and the deceleration values.

Table 4.8: Examples of simulation results. The conditions of each run are presented in Table 4.7

Run number	Gap with the reference average speed (%)	Gap with the reference fuel consumption (%)	<i>RDI</i>
1	0	0	2.886
2	+16.5	-17.9	4.094
3	-5.8	-18.3	3.326
4	-2.8	-2.8	2.886
5	-19.3	-17.8	2.836

between the tested driving features and their corresponding *RDI*. As it can be seen from that table, average deceleration shows a strong correlation with *RDI*, displaying a Pearson's correlation coefficient as high as 0.79, when fixing all other driving features, *i.e.*, cruising speed and crossing speed. Furthermore, as illustrated in Table 4.8, we discover that the lower the absolute value of average deceleration, the higher the *RDI*. For example, run 3 displays a small deceleration and depicts a high *RDI*. These results thus confirm that using coasting, *i.e.*, having a lower deceleration during the braking phase, is an important driving feature of rational driving.

Table 4.9: Correlations between the tested driving features and their *RDI* measured in terms of Pearson's coefficient

Varying driving feature	Fixed driving features	Average correlation with <i>RDI</i>
Average deceleration	Cruising speed, Crossing speed	0.79
Crossing speed	Average deceleration, Cruising speed	0.98

4.5.3.2 Impact of the crossing speed

For a fixed couple of average deceleration and cruising speed, the only varying parameter is the crossing speed. As it can be seen from Table 4.9, as compared to average deceleration, crossing speed displays an even higher Pearson's correlation coefficient with *RDI*, while fixing the average deceleration and cruising speed. Furthermore, as illustrated in Table 4.8, the experimental results show that the higher the crossing speed on the roundabout, the better the *RDI*. These simulation results further confirm the correlation between crossing speed and rational driving that we discover using real driving data.

4.5.3.3 Summary

For each of the two driving features tested in simulation, namely average deceleration and crossing speed, the correlation with *RDI* is very high. This conclusion holds for three different cruising speeds tested in simulation. The simulations enable us to isolate and quantify the impact of one driving feature with respect to rational driving while fixing all the other ones. The simulation results further confirm the global trends that we observe on real driving data.

4.6 Simulation-based impact study of driving features when approaching a toll

Section 4.4.2 demonstrates that one of the most differentiating driving features in terms of rational driving on highway road is coasting. This is promising since coasting is related to one of the main rules of rational driving. Moreover, tolls are one of the key events on a highway road, due to the few kinds of events that can take place on highways. In this section, we aim to provide a quantification of the impact of coasting when approaching a toll using simulations, with the same methodology as the previous section concerning roundabouts.

4.6.1 Simulating driver behaviors on the simulated toll

The input of the simulation tool will be the digital representation of the highway section. Fig. 4.19 displays a Google Earth view of the selected toll. Fig. 4.20 presents the simulated signals corresponding to this section: a speed target at 90 km/h during 4.5km and a final stop, with a zero slope signal.



Figure 4.19: Google earth view of the toll

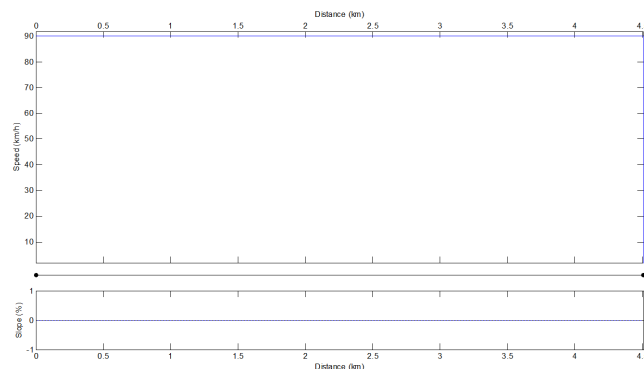


Figure 4.20: Digital representation of the highway section. At the top, there is the speed target at 90 km/h *vs.* distance signal. In the middle, the black points represent the stops. At the bottom, this is the signal of the slope *vs.* distance.

4.6.2 Impact of the percentage of coasting on RDI

Once the file representative of the highway section is created, we can estimate the impact of coasting during the toll approach. We propose to study three different behaviors:

- One behavior without coasting, *e.g.*, representative of a non-efficient rational driver.

- One behavior with 54% of the distance of the event spent in coasting, *e.g.*, representative of a medium rational driver.
- One behavior 75% of the distance of the event spent in coasting, *e.g.*, representative of an efficient rational driver.

These percentages represent the average percentages of coasting when approaching a toll for medium and non-efficient drivers from the database. The results are presented in Table 4.10. As expected the RDI is the highest for the behavior with the maximum amount of coasting, and is the lowest when there is no coasting. Therefore, the Pearson's coefficient between RDI and percentage of coasting distance is 0.89. This result further confirms the global trend that we observe using real driving data.

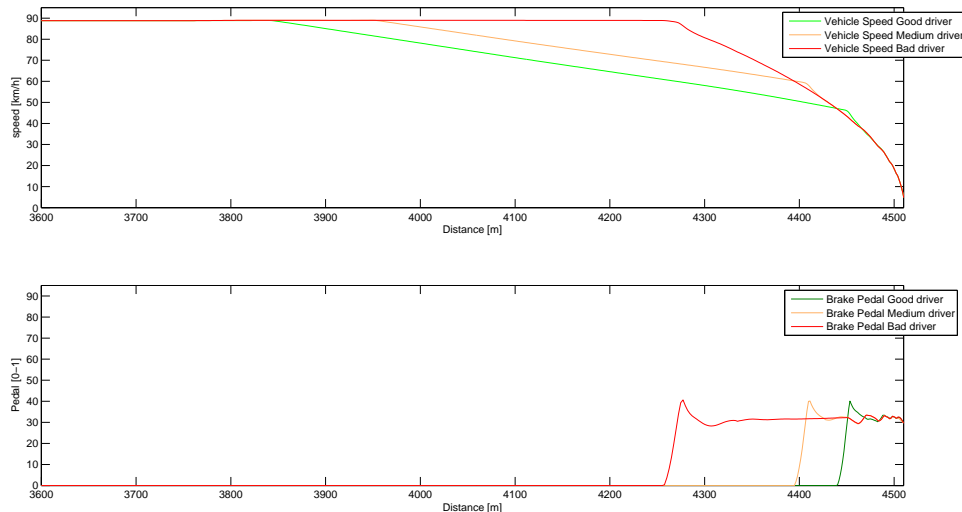


Figure 4.21: Digital representation of the highway section

4.7 RDI correlation to a cost model

The study presented in the previous paragraphs is based on the RDI, an index that we introduced to measure the rational driving ability of each driver. We compare this index to a cost model for this specific vehicle to prove that the RDI is relevant with the concept of rational driving. The RDI does not need vehicle costs information and is consequently easily calculated whatever the vehicle. Conversely, a cost model is specific to a given vehicle and therefore cannot be generalized as a rational driving

Table 4.10: Examples of simulation results for three different coasting behaviors

Driver type	Gap with the medium driver average speed (%)	Gap with the medium driver fuel consumption (%)	Gap with the medium driver <i>RDI</i> (%)
Efficient driver	-2.32	-11.47	+10.33
Medium driver	0	0	0
Non-efficient driver	+2.39	+32.46	-22.75

indicator. However, in the specific case we are looking at, we aim to calculate a cost model to validate the relevancy of the RDI. Thus, we implement a freight tariff calculation that takes into account the benefit of fuel economy balanced with the cost of the loss of time. This index evaluates in € the cost of the average speed and the average fuel consumption. The freight tariff represents the cost of transportation per ton per kilometer. The objective for transportation companies is to minimize this cost, whilst maximizing RDI.

For this D-wide 280hp vehicle the freight tariff (FT) is obtained with the following formula:

$$FT \left(\frac{\text{€}}{T.km} \right) = \frac{\text{Transportation Cost (€)}}{\text{Transportation work (T.km)}} \quad (6)$$

When replacing the transportation costs and the transportation work with the corresponding values, the previous equation becomes:

$$FT = \frac{A}{\text{Speed}} + B * \text{Fuel Consumption} \quad (7)$$

where *Speed* is the average speed in *km/h* on the entire path, *Fuel consumption* is the average fuel consumption in *L/100km* on the entire path, and A and B are constants depending on the cost of such a vehicle. The cost comprises: acquisition, insurance, taxes, load carrier, maintenance, tires, and so forth. The average costs are calculated from a truck rental agency prices, *i.e.*, 36,745€ per year for a typical delivery vehicle. The price of the fuel comes from International Energy Agency, *Energy Prices and Taxes* report published in May, 2013, and is representative for the EU.

For each roundabout simulation run, we compute the FT with the corresponding fuel consumption and average speed. If we compare the runs with the same cruising speed but different crossing speeds as shown in Fig. 4.22, we observe that the higher the crossing speed in roundabouts, the higher the RDI and the lower the freight tariff. This result was expected, and shows that having a higher speed in roundabouts has a positive effect both on the fuel consumption and the average speed. However, the conclusion is less obvious if we compare different cruising

speed. The RDI is higher with a lower cruising speed (for example between 80 and 70 km/h the RDI goes from 2.42 to 2.58 for a crossing speed at 24km/h) but the freight tariff increases. When the cruising speed is even lower (60 km/h instead of 70km/h) the RDI is almost equal (2.596 instead of 2.58 *i.e.* +0.6%) but the freight tariff is much higher (+4.9%). The same conclusion goes for the impact of a lower absolute value of deceleration, as presented in Fig. 4.23. In these examples, the lower the absolute value of the deceleration, the lower the RDI on average, but the freight tariff decreases too.

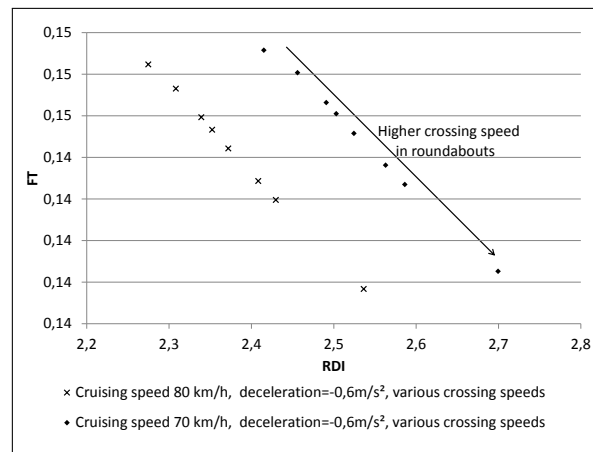


Figure 4.22: FT (€ per ton.km) vs. RDI values for different runs. For higher crossing speeds in roundabout, the RDI increases and the FT decreases.

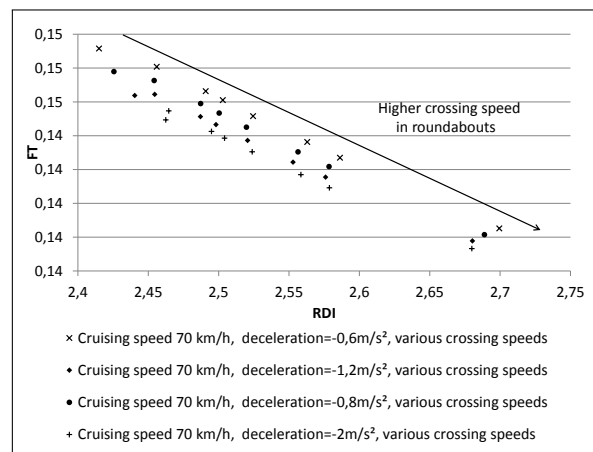


Figure 4.23: FT (€ per ton.km) vs. RDI values for different runs. For lower absolute value of the deceleration, the RDI decreases on average, but the FT decreases too.

The global Pearson's correlation coefficient between RDI and FT is -0.73. The RDI then has a global trend correlated to the freight tariff but is not highly correlated concerning the specific impact of cruising speed and average deceleration for braking. Actually, the impact of only one parameter is difficult to isolate when the parameter does not imply a positive effect on both the fuel consumption and the average speed. However, by combining several driving features representative of rational driving behaviors, the impact is clearer and can thus be useful to transportation companies. For instance, as can be seen in Table 4.11, an efficient driver who crosses roundabouts quickly and who has a smoother deceleration can have a freight tariff 2.1% lower than a non-efficient driver who has low crossing speeds in roundabouts and who brakes aggressively.

Table 4.11: Differences between RDI and FT for an efficient and a non-efficient driver with a reference driver.

Cruising speed	Crossing speed	Deceleration	RDI	FT
80	20	-0.8	Reference	Reference
80	17	-2	-1.3%	+0.5%
80	24	-0.6	+2.8%	-1.6%

FT is an opportunity to evaluate the relevancy of the RDI. FT and RDI give the same conclusions concerning driving parameters which positively impact both fuel consumption and average speed. However, when the impact of the driving features is smaller, the FT is not always lower when the RDI is higher. That is why better results are obtained when several driving features vary instead of only one.

4.8 Conclusion

In this chapter, we aimed to concretely find driving parameters that are correlated to rational driving. Therefore, we first introduce an indicator, namely *RDI*, for rational driving index, which enables us to quantify how good a driving behavior is on a given travel path. We then studied various driving features related to rational driving, using real driving data collected from 34 drivers who piloted the same truck on an extra-urban road and on a highway road. We measured the correlation between driving features and rational driving using real driving data and also in simulations. We focused in particular on the use of coasting and the crossing speed at roundabouts, and the use of coasting when approaching a toll.

The correlations between the driving features and the RDI were found very high in simulation, and further confirms what is intuitively expected by knowing the rules of rational driving. Finally, we also compared the RDI to a cost model for this specific vehicle, to prove that the RDI is relevant to evaluate rational driving

capacity.

At this stage of the study, we can recognize driving events with driving data and we have correlated driving features to rational driving on two different events. The next chapter will present the implementation of these results in our testing tools and the overall structure of the proposed system to create different driving behaviors in terms of rational driving.

Driver model generator

In the preceding chapters, we analyzed a way to automatically detect driving events, then we found some differences between driver behaviors when faced to different driving events. At this stage we have not dealt with the implementation of these results into a framework which is able to model these different behaviors.

We presented several examples of driver model implementation using different methods in Chapter 2. Some tune the control system parameters to obtain a different response to the same inputs. Some tune driving parameters, and some tune the setpoints, which imply different inputs to the lateral and/or longitudinal controllers depending on the desired driver behavior. However, they are not comparable with our study, because their purpose is different, *e.g.*, the COSMODRIVE model, or since they are not based on an analysis of real driving data, *e.g.*, the ACEA model, IPG model. The challenges in our work are to create driver models based on real driving data, but directly implementable in our existing tools and rational-driving centered. Chapter 3 and 4 focused on rational driving behaviors from real driving data and the adaptation of our results to digital roads. We now present the choices made for the implementation of different driving styles in our internal testing tools.

5.1 Cycle Tuning Structure

The previous chapters suggest that the differences between drivers when they have to drive on the same road lie in the way they control their vehicle when facing the events and the choice of vehicle speeds. Thus we aim to build a model which adapts a road (*i.e.*, a cycle) to a desired driver type which is itself relative to an initial driver type. This is because we mainly observed relative gaps between drivers, rather than absolute ones. For example, the crossing speed of a roundabout for an efficient driver is computed relative to the crossing speed of a medium or non-efficient driver. Table 5.1 presents our choices of practical applications concerning the three levels of driving activity adapted to our goal.

Table 5.1: Our choices in the three levels of driver models

Real driver	Virtual driver
Strategic decisions: road planning	Preprocessing of the road file
Tactical decisions: maneuvering	Choice of speed and actuators control
Operational decisions: execution tasks	PID controllers

Fig. 5.1 presents our driver model structure. The gray cells are existing tools which are not part of the study but are necessary to understand the method. The blue and orange cells are within the scope of this study. Here are the following steps of the method:

- Create a road, either via a cycle generator tool (left gray cell), or by using an existing road file.
- Recognize the driving events (upper blue cell) by using the decision trees learnt in Chapter 2.
- Recognize the vehicle dynamics and the driver efficiency management of the original cycle (middle and bottom blue cells). Although these functions are not part of this current work, they allow us to have reference vehicle and driver, and the following steps are done relatively to these references.
- Self-learning of the target vehicle (light orange cell). The objective is to build target vehicle acceleration maps.
- Adapt the original cycle to the target vehicle and the desired driver type (orange cell). This is the core of the structure, and is based on the conclusions of Chapter 3. The new targets (*e.g.*, speed, actuators positions) are computed relative to the targets from vehicle/driver references.
- Use this adapted cycle in the existing tools for simulation, or on chassis-dynamometer (right gray cell) to obtain fuel consumption and average speed results. In this chapter we present the results provided with our simulation tool. The tests on chassis-dynamometer are ongoing and will be the subject of an internal technical report.

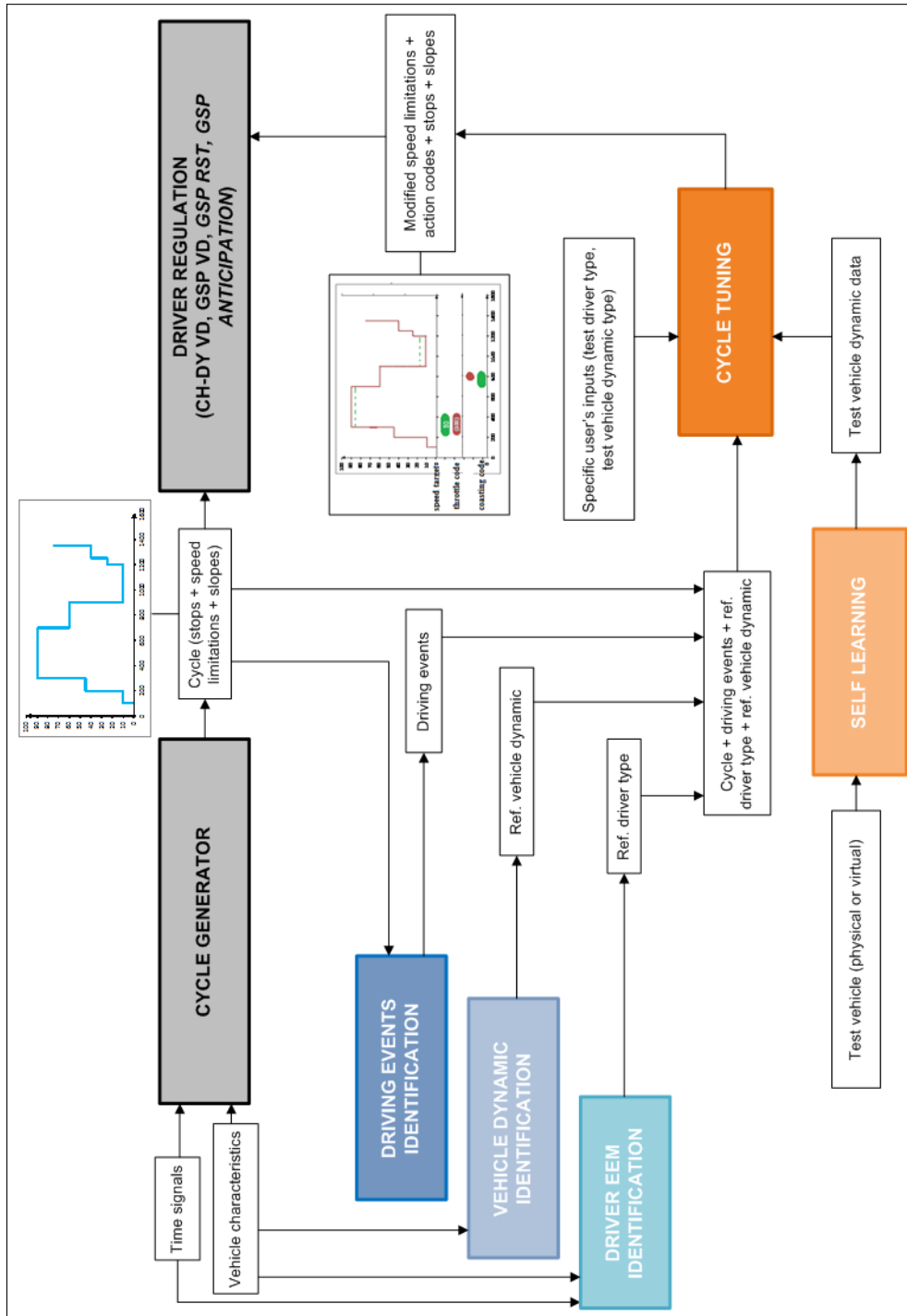


Figure 5.1: Structure of the driver model generator.

5.1.1 Driving events identification

The first step of the process is driving events recognition on the selected cycle. Event identification inputs are cycle information: speeds limits, stops, slopes, and road type. This block must recognize all the driving events presented in Chapter 3. We saw in Chapter 3 that events classification rates range from 80% to 90% on digital roads.

Events identification is implemented in the following manner with Matlab functions: the function *find events* browses the cycle and launches the *identify event* function for each speed change. The function *identify event* works as indicated below:

- Identify type of events, *i.e.*, estimate if it is a *Throttle=0* or *Throttle>0* event. This step is necessary because we do not have information about actuators, since we are only using target speeds. Two rules were created for this identification:
 - If the final speed is in the range $[Speed_{initial} - 10km/h; Speed_{initial} + 10km/h]$, the event type depends on the slope. For a positive slope, the event will be of type *Throttle>0*, while for a negative slope the event will be of type *Throttle=0*. For example, if the vehicle is on a road with a significant negative slope and the vehicle does not accelerate greatly, the driver will intentionally slow the vehicle down using brakes or retarders: this is then a *Throttle=0* event.
 - If the final speed is outside this range, either the final speed is superior to the initial speed, in which case it is a type *Throttle>0* event, or conversely it is a type *Throttle=0* event.
- Identify the event by using the adapted classification method. It means that we apply the decision tree built with the real database for each type of event (*Throttle=0* or *Throttle>0*). However, the input features are those that we can compute on the simulation data, namely maximum and minimum speed, Δ speed, road type, and stop.

Two methods are used for events identification: Matlab decision trees (with an Excel file database) and WEKA decision tree written in Matlab code. The WEKA decision trees are those obtained in Chapter 2. The Matlab decision trees are not used for now, but they can be useful to adapt the method quickly if the database is expanded. Finally, the event list is stored in the cycle file, to avoid the re-identification of the events later in the process.

As we have seen in Chapter 3, the classification rate for urban, main road and highway databases are comprised between 80% and 90%. Similar to the work with real data, the confusion matrices only show errors between situations which could be considered as similar driving behaviors.

5.1.2 Driver efficiency energy management (EEM) and vehicle dynamic identification

Once the events are identified, the next step is the identification of driver type (EEM) and of the type of vehicle corresponding to the current cycle, see Fig. 5.1. The objective of the functions *find driver type* and *find vehicle type* is to label the reference driver and vehicle, with the help of data from the cycle and/or from the user. Estimating the triplet (*reference cycle*, *reference vehicle*, *reference driver*) is necessary to create the triplet (*desired cycle*, *desired vehicle*, *desired driver*) as we work relative to the reference values. These three objects are inseparable.

We will use high-level parameters to identify the driver type, such as the number of brake events per 100 km. These parameters will depend on the truck usage. Several parameters have been tested, but for one usage each time, and with data from one path only. We will not present the results of the high-level studies to identify the driver type as it is not in the scope of this thesis. However, we can say that for driver type identification we use common driving parameters from state-of-the-art, and we aim at covering a wide range of truck utilization, such as long-haul, distribution, refuse, and construction.

The number of driver types is set to three different behaviors. For now the reference driver type is imposed as a medium driver. The three classes are efficient, medium and non-efficient drivers, respectively. We could differentiate driver types more accurately, *i.e.*, weighted by speed, such as *efficient but slow driver*, or *efficient and fast driver*. However, using three classes appears here to be the best compromise between the size of our database and the target number of driver types for simulation and chassis-dynamometer. For example, having too many driving behaviors to test may require too much time during the development process. We will also assume that a driver has the same behavior along the whole length of the road. For example, the driver does not switch from efficient to non-efficient while driving.¹ Moreover, as we have seen in related works (see Section 2.2.3), authors often distinguish between two and five classes.

For now the vehicle dynamics are either imposed, or the weight to power ratio is used. Ratios will be ranked by class, however the limits of the classes are not defined yet.

To anticipate future needs, these two functions, namely driver identification, and vehicle identification, are already in the *Cycle generator* tool and their inputs include some vehicle parameters to identify more precisely the vehicle dynamics.

5.1.3 Cycle tuning

The *cycle tuning* block is the core of the concept, as presented in Fig. 5.1. Fig.5.2 focuses on the cycle tuning bloc and illustrates its principle. The inputs are all

¹Actually, switching the driver type during the cycle is quickly implementable since we preprocess the cycle. However, it was not the goal of this work.

the outputs of the preceding functions: existing cycle (stops, speed limitations, slopes and road type), identified events, vehicle dynamics reference and identified driver type reference. The *cycle tuning* function adapts the speed limits relative to these references and to desired driver and desired vehicle. The function also defines actuator codes to activate actuators on imposed positions.

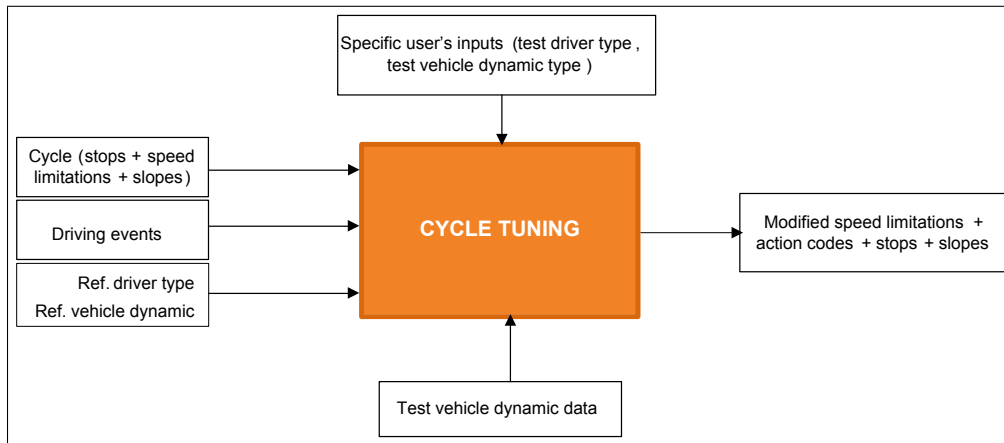


Figure 5.2: Cycle tuning function principle.

Our initial idea was to create a continuous speed target signal *vs.* distance. As interesting as this idea was, it was not really adapted to our testing method. The reason being that we always test vehicles relative to others, and because of the differences between vehicles, *e.g.*, gear changing, we would not have obtained comparable results while following the same original cycle. Moreover, the regulation system is not adapted yet to follow a continuous speed target, especially at low speeds. Consequently we chose to tune only vehicle-independent parameters, *i.e.*, speed limits, and actuators code.

Actuator codes are setpoints concerning actuators: they prevail over target speed setpoint until the vehicle reaches the speed limit.

There are two codes:

- The throttle code sets a throttle pedal position (in %). The code is defined by a starting distance and the percentage of throttle position. The virtual driver exits this operating mode at the next speed target distance or when the throttle position given by the PID control system becomes lower than the imposed throttle pedal position. That means that the speed target is close to the current speed, therefore the virtual driver must stop using the imposed throttle pedal position to adjust to the final target speed with the regular PID controller.
- The coasting code sets brake, throttle, and retarders to zero. The code is only defined by a starting distance. Coasting happens only before braking

and stopping phases. The virtual driver exits this operating mode when the current distance matches the next corresponding braking distance (this braking distance is computed with the average braking deceleration, which is a parameter defined by the user).

5.1.4 Self-learning cycle

In addition to these modifications on the output cycle generated by the *cycle tuning* function, we also introduce a self-learning cycle in order to adapt the *cycle tuning* function's output to the vehicle more precisely. The self-learning cycle makes it possible to learn vehicle data, such as coasting deceleration and traction acceleration. While it is possible to estimate coasting theoretically, the traction accelerations depend on gear changing strategies and on functions such as acceleration limiters, therefore learning the vehicle dynamic behavior is necessary to adapt the cycle in the best possible way to the target vehicle.

The self-learning cycle can be done on simulation or on chassis-dynamometer, and must be executed with the target vehicle (real or simulated) before using the *cycle tuning* function. The output of the learning cycle is processed in the *cycle tuning* function in order to build maps of longitudinal acceleration corresponding to the vehicle. More precisely, a self-learning cycle is a succession of acceleration plus braking/coasting phases as illustrated in Fig. 5.3 and in Fig. 5.4.

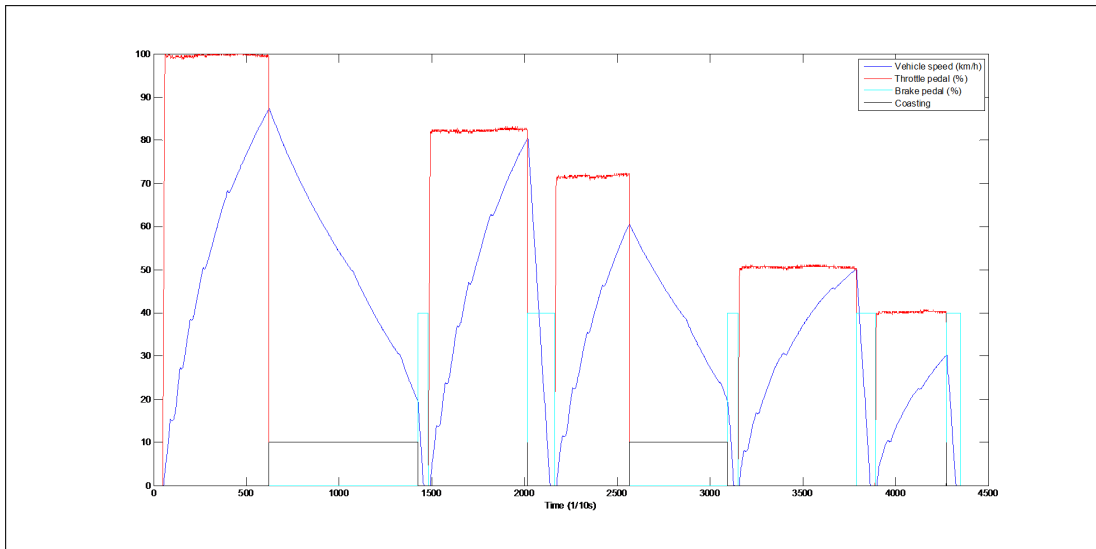


Figure 5.3: Example of a self-learning cycle on a chassis-dynamometer (vehicle loaded to 14T - slope 0%)

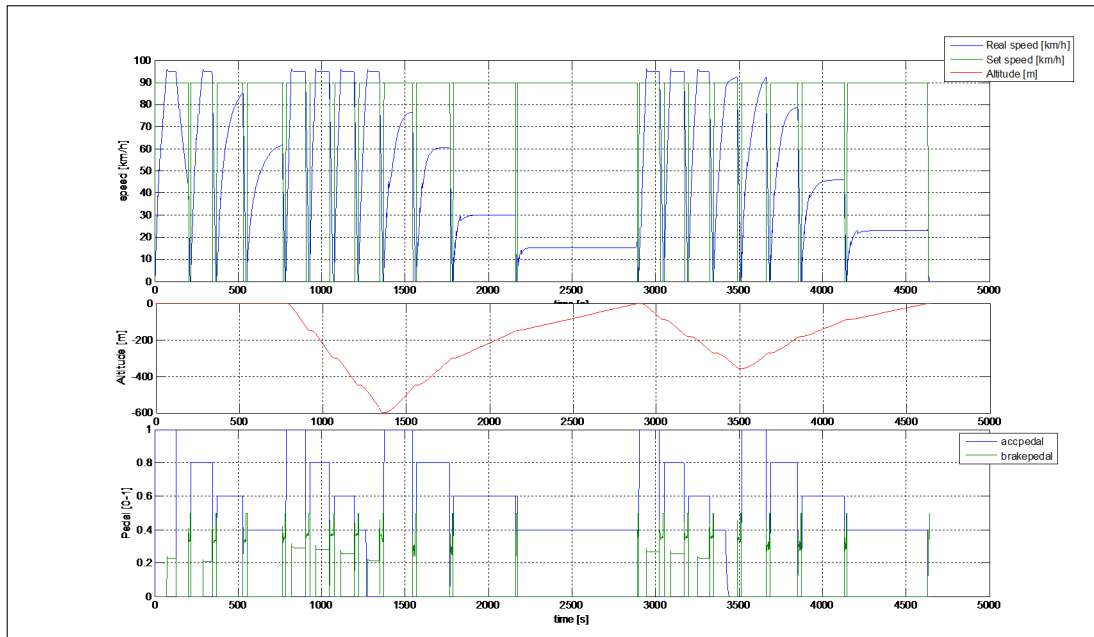


Figure 5.4: Example of a self-learning cycle in a simulation (vehicle loaded to 14T)

The user defines the different target speeds, the percentages of throttle pedal to reach these speeds, and if there is coasting after each phase. There are five or less phases during the self-learning cycle. Then the user can run the self-learning cycle on different slopes depending on the need and the allowed time. When the slope is not equal to 0, there is no need to do coasting phases, since the coasting deceleration can be extrapolated on non-zero slopes from the coasting deceleration values on the zero slope. The objectives of the self-learning cycle are to learn deceleration values during coasting, and to estimate traction acceleration for the output cycle during the acceleration phases. The coasting deceleration is used before braking phases, *e.g.*, when approaching a roundabout to differentiate driver behaviors. The traction acceleration can be used during traction phases, such as take-offs.²

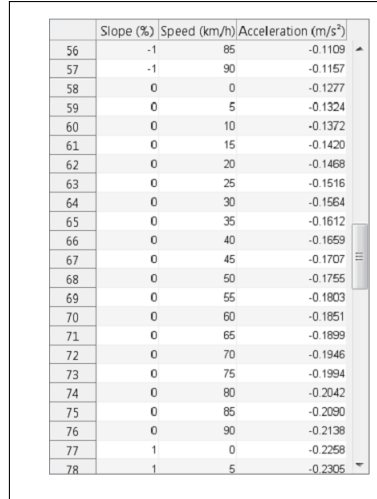
5.1.4.1 Coasting deceleration map learning

During the coasting phases of the self-learning cycle, we can estimate the deceleration of the vehicle with Newton's Second Law by using the engine friction torque (depending on engine speed), the rolling resistance, the drag force, and the weight component parallel to the slope.

For each range of speed reached during the self-learning cycle we then obtain a coasting deceleration value for different slopes (*e.g.*, -5% ; -3% ; -1% ; 0% ; +1% ; +3% ; +5%). Then a linear extrapolation computes the coasting deceleration to

²However, driving analysis concerning take-offs is ongoing and will be the subject of an internal technical report

other speeds to cover the range $[0; 90\text{km/h}]$ with 5km/h intervals. The final result is a table $[\text{slope}, \text{speed}, \text{coasting deceleration}]$ as shown in Fig. 5.5. This table will be used when processing the cycle to adapt it to different driver behaviors.



	Slope (%)	Speed (km/h)	Acceleration (m/s ²)
56	-1	85	-0.1109
57	-1	90	-0.1157
58	0	0	-0.1277
59	0	5	-0.1324
60	0	10	-0.1372
61	0	15	-0.1420
62	0	20	-0.1468
63	0	25	-0.1516
64	0	30	-0.1564
65	0	35	-0.1612
66	0	40	-0.1659
67	0	45	-0.1707
68	0	50	-0.1755
69	0	55	-0.1803
70	0	60	-0.1851
71	0	65	-0.1899
72	0	70	-0.1946
73	0	75	-0.1994
74	0	80	-0.2042
75	0	85	-0.2090
76	0	90	-0.2138
77	1	0	-0.2259
78	1	5	-0.2305

Figure 5.5: Example of a table of coasting deceleration (vehicle loaded to 14T - slope 0%)

5.1.4.2 Traction acceleration map learning

During the acceleration phases we can measure the acceleration of the vehicle depending on the current slope, speed and throttle pedal position. We also extrapolate the results to cover the widest range of speeds and slopes possible. The output table will be used to give a throttle position setpoint depending on driver behavior, *i.e.*, on the desired acceleration.

For each acceleration phase, we smooth the signal of acceleration *vs.* speed of the vehicle. This is because we cannot use the raw measured values since the gear changes add zero crossings to the acceleration signal, as we can see on Fig. 5.6. Then we fit the filtered signal by a model to extrapolate the acceleration values from 0 to 90 km/h even if the covered range of speed is lower during the acceleration phase. Since the acceleration is always positive, and tends to decrease and to stabilize with higher speeds, we chose an exponential model:

$$y = a * \exp(b * x)$$

where a and b are the parameters of the model, x is the input of the model, and y is its output. This model produces good results with an average RMSD of 0.049 and an average correlation coefficient R^2 equal to 0.89. Then we build a table by subsampling this fitting result from 0 to 90 km/h with a step of 5km/h , for each

slope and each throttle pedal position existing in the self-learning cycle. The final result is a table [*slope*, *speed*, *% throttle pedal*, *acceleration*].

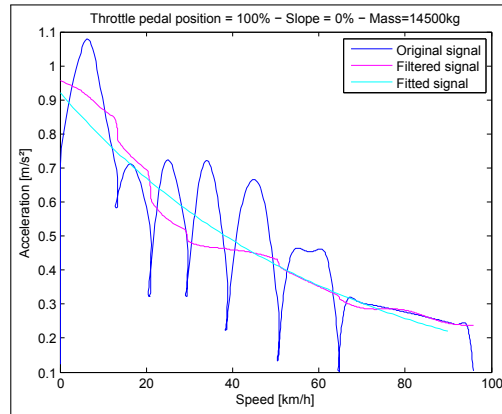


Figure 5.6: Example of acceleration *vs.* speed signal acquired during a self-learning cycle (blue), filtered acceleration signal (magenta) and signal fitted by an exponential model (cyan).

Thus, the self-learning cycle allows us to build an acceleration map when coasting and when pressing on the throttle pedal. By extrapolation, these tables cover all the cases encountered on the cycles.

5.1.5 Chassis-dynamometer and simulation controllers

The output of all previous steps is a cycle adapted to the desired driver behavior and desired vehicle dynamics. This cycle can be used as an input to the chassis-dynamometer or simulation controllers that will follow the new speed and actuator targets. These test facilities were presented in Section 1.2.1.

5.2 Implementation

The *cycle tuning* function is implemented in Matlab. The user interface is presented in Fig. 5.7. This program enables us to:

- Build the acceleration tables with the self-learning files as inputs. These functions are activated when pushing the two buttons on the left side, as can be seen in Fig. 5.7.
- Choose a cycle. The function is activated when pushing the top button on the right side of the interface.
- Select a desired driver type (efficient, medium, or non-efficient). The choice is made with the buttons in the middle of the right side of the interface.
- Select a desired vehicle type (2 as default value). The choice is made with the

buttons in the middle of the right side of the interface.

- Compute the new cycle adapted to the desired driver type and vehicle. This function is activated when pushing the bottom button on the right side of the interface.

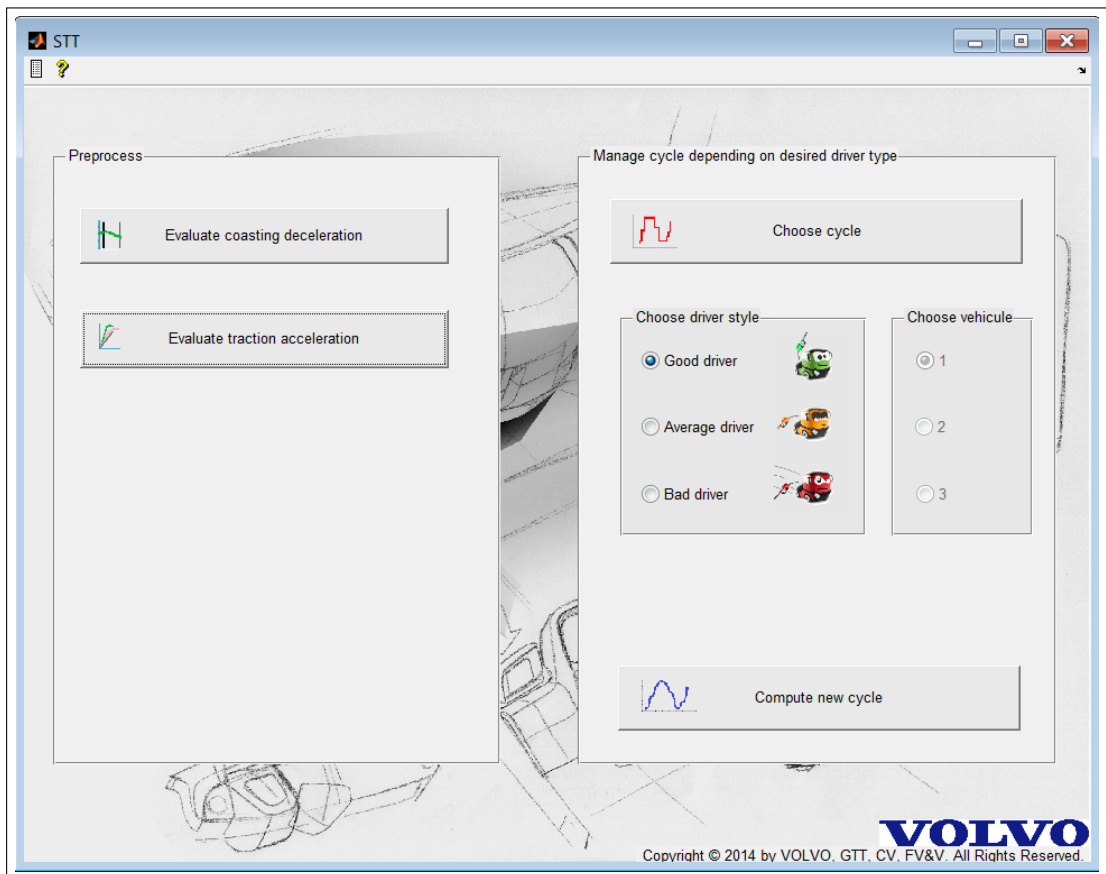


Figure 5.7: View of the user interface of *cycle tuning*

The main function goes through the list of events and for those where we have identified differences between drivers' behavior, we adapt the cycle to the desired behavior. According to the conclusions of Chapter 3, we have modified three event classes so far:

- Roundabout: the final speed is modified depending on reference and desired driver type. Coasting is added for efficient and medium drivers.
- Toll: coasting is added for efficient and medium drivers.

- Stops: the stop time for efficient and medium drivers is reduced, and coasting is added accordingly.³

5.3 Results

We present here the results obtained for each specific situation, namely roundabout, toll, and stop.

5.3.1 Roundabout

We observed that efficient drivers cross roundabouts at higher speeds and use more coasting than other drivers. We implement a linear model of the crossing speed depending on the desired and the reference driver type. The model is built with the observed real crossing speeds. For example, the crossing speed for an efficient driver type compared to the crossing speed of a non-efficient driver type is: $V_{eff} = 0.4516 * V_{non\ eff} + 15.89$. The principle of implementation is provided by Algorithm 3. We use the parameter of braking deceleration, the average percentages of coasting distance observed from the real roundabout data for the three kinds of driver behaviors, and the coasting deceleration corresponding to the current speed, and the average slope. We assume that braking is always achieved with constant deceleration.

Fig. 5.8 and Table 5.2 show the results in terms of fuel consumption, average speed, and RDI for three different behaviors representative of efficient, medium, and non-efficient drivers. In Fig. 5.8, the top graph is the speed (km/h) *vs.* distance (m) signals for an efficient driver (green), a medium driver (orange), and a non-efficient driver (red). The bottom graph represents the brake and throttle pedals position (%) *vs.* distance for an efficient driver (blue and green, respectively), and the brake pedal position (%) for medium driver (orange) and non-efficient driver (red). As expected, the efficient driver's fuel consumption is the lowest with the highest average speed. As opposed to a non-efficient driver who has the highest fuel consumption with the lowest average speed. The gaps between drivers are close in terms of absolute value. However, these results only quantify the impact on one roundabout, whereas there are many events on a full cycle, and the impact of each event is added to one another. Thus, the global impact on a full cycle can be much more significant.

5.3.2 Toll

The second event where we obtain quantifiable differences between drivers is the toll. The principle is the same as for the roundabout except that the final speed does

³We did not present the theoretical results for stops in the previous chapter as they are not based on a correlation with the RDI. They are based on average values of stopping time for the different driver types, as presented in Section 5.3.3

Algorithm 3 Roundabout approach implementation

Require : Dr_{ref} - Reference driver type, Dr - Desired driver type,
 V_i - Initial speed of the event, V_f - Actual final speed of the event, *i.e.*, roundabout crossing speed,
 d_f - final distance of the event,
 a_{Brake} - the braking deceleration,
 $slope$ - slope of the event,
 $Table_{Coasting}$ - the coasting acceleration table learnt with the self-learning,
 $\%Coasting$ - the percentage of coasting during roundabout approach corresponding to the desired driver type.

Ensure : $Dist$ - Start coasting distance,
 V_{fD} - Roundabout crossing speed corresponding to the desired driver type.

if $Dr_{ref} \neq Dr$ **then**
 $V_{fD} = f(V_f, Dr_{ref}, Dr)$ where f is the linear model.
End distance of coasting $Dist_{End\ Coasting} = d_f - \%Coasting * (d_f - d_{end})$, where
 $(d_{end} = \frac{(V_{fD}^2 - V_i^2)}{2 * a_{Brake}})$
Speed corresponding to the end distance of coasting $V_{Coasting} =$
 $\sqrt{V_{fD}^2 - 2 * a_{Brake} * (d_f - Dist_{End\ Coasting})}$
 $a_{Coasting} =$ interpolation of $Table_{Coasting}$ at the point $(V_i, slope)$.
 $Dist = Dist_{End\ Coasting} - (\frac{(V_{Coasting}^2 - V_i^2)}{2 * a_{Coasting}})$

return $Dist, V_{fD}$

Table 5.2: Simulation results of different roundabout approach behaviors.

Driver type	Non efficient	Medium	Efficient
Start coasting distance (m)	∅	1320	1348
Crossing speed in roundabout (km/h)	15	19	24.5
Fuel consumption (L/100km)	27.5	26.1	26.0
Average speed (km/h)	60.1	61.1	62.0
RDI	2.185	2.341	2.386

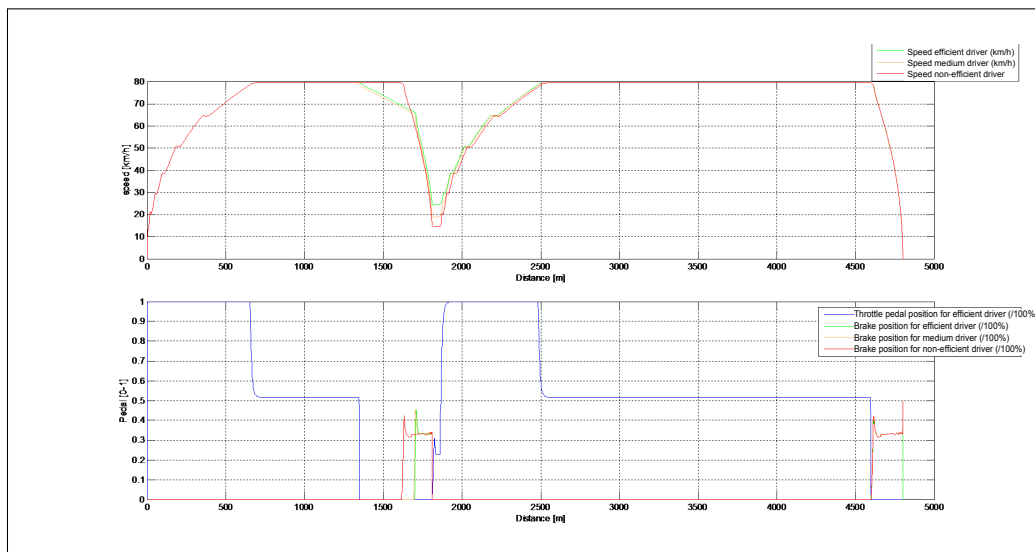


Figure 5.8: Simulation results of different roundabout approach behaviors.

not change, and the coasting percentage values are different. Unlike the roundabout approach, the efficient driver has the lowest average speed when approaching a toll. However, the fuel savings are so high that the RDI is better for an efficient driver. Non-efficient drivers have the lowest RDI because of the high fuel consumption, even if the average speed is higher.

Table 5.3: Simulation results of different toll approach behaviors.

Driver type	Average speed (km/h)	Fuel consumption (L/100km)	RDI
Non efficient	72.20	28.69	2.52
Medium	71.22	25.91	2.75
Efficient	70.04	24.95	2.81

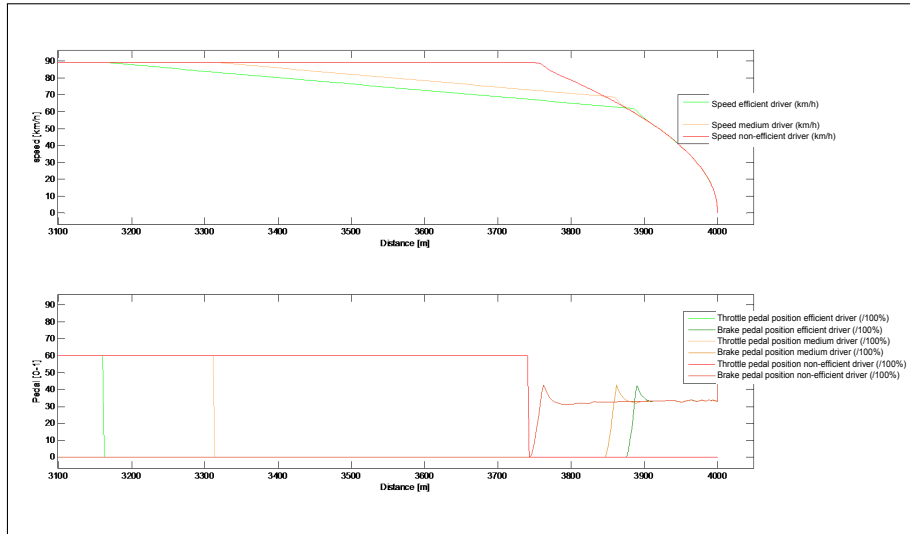


Figure 5.9: Simulation results of different toll approach behaviors.

5.3.3 Stops

In order to cover a wider range of situations, we also choose to implement different driving behaviors on a frequent urban situation. This is because an urban situation is complementary to roundabouts which are representative of an extra-urban area, and to tolls that are key events on a highway. Consequently we study the time spent at the stops in a urban section.

First, we observe that the probability of stopping at a red traffic light is similar for efficient, medium and non-efficient drivers. This is due to the random probability of a red traffic light, even if efficient drivers try to avoid them by anticipating. The density of stops for each type of driver on the urban section is presented in Table 5.4. However, since efficient drivers anticipate the traffic light color, they spend less time at stop, as presented in the 5th line of the same table. In a cycle, a stop is defined by the distance of the stop and the time spend at the stop. Therefore, we can reduce the time spent at a stop for efficient and medium drivers, and add the corresponding coasting distance during the approach of the stop, to balance the time saved at stop. This is because efficient drivers spend less time at a stop, and a red light lasts the same time for all drivers. Consequently, efficient drivers arrive later at the stop distance. Since the initial speed is the same for all drivers during tests, they must have a lower average deceleration to reach the stop point, and they therefore use more coasting.

The principle of implementation of stop approach is presented in Algorithm 4. The first output is the stop time for the target driver, which is the difference between the reference stop time and the average time saved by efficient and medium driver compared to a non-efficient driver. The second output is the start coasting distance, which is calculated with the coasting deceleration interpolated in the coasting accel-

Table 5.4: Stop features in urban areas.

Driver type	Non efficient	Medium	Efficient
Total stop time (s)	58.95	46.85	43.96
RDI	2.03	2.2	2.42
Density of stops per km	1.11	1.0	1.07
Average stop time (per stop)	19.08	16.73	14.65

eration table learnt using the self-learning cycle. Fig. 5.10 presents the differences between a reference and a target cycle: the stop times are represented by the stems, and the coasting is represented by the blue rectangles. It can be seen that the coasting distance is different depending on the initial speed. Table 5.5 shows the results on a urban cycle which includes many stops. The cycle, which is representative of an efficient driver, has a higher average speed (19.47km/h instead of 18.98km/h for a non-efficient driver) and a lower fuel consumption (28.59L/100km instead of 30.05L/100km), and therefore a better RDI.

Algorithm 4 Stop approach implementation

Require : Dr_{ref} - Reference driver type, Dr - Desired driver type,

V_i - Initial speed of the event,

d_f - Final distance of the event,

t - Stop time, t_{Driver} - Time saved by desired driver type during stop,

a_{brake} - Braking deceleration,

$slope$ - Slope of the event,

$Table_{Coasting}$ - Coasting acceleration table learnt with the self-learning.

Ensure : $Dist$ - Start coasting distance,

$StopTime_{target}$ - Stop time corresponding to the desired driver type.

if $Dr_{ref} \neq Dr$ **then**

$$StopTime_{Target} = StopTime_{Reference} - t_{Driver}$$

$$BrakingTime_{Target} = BrakingTime_{Reference} - t_{Driver} \text{ where}$$

$$BrakingTime_{Reference} = \frac{-V_i}{a_{brake}}$$

$$\text{Speed at the end coasting distance } V_{Coasting} = -BrakingTime_{Target} * a_{Brake}$$

$$\text{End distance of coasting } Dist_{End\ Coasting} = d_f - \frac{-V_{Coasting}^2}{2*a_{Brake}}$$

$$a_{Coasting} = \text{interpolation of } Table_{Coasting} \text{ at the point } (V_i, slope).$$

$$Dist = Dist_{End\ Coasting} - \left(\frac{(V_{Coasting}^2 - V_i^2)}{2*a_{Coasting}} \right)$$

return $Dist, StopTime_{target}$

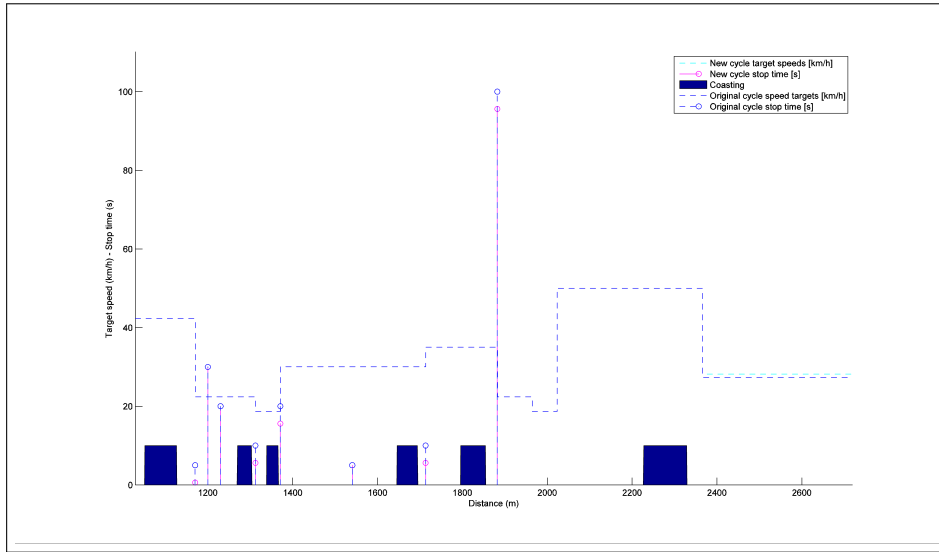


Figure 5.10: Example of stop implementation.

Table 5.5: Results of an Urban cycle.

Driver type	Non efficient	Efficient
Total stop time (s)	1322.8	1207.1
RDI	0.63	0.68
Average speed (km/h)	18.98	19.47
Average fc (L/(100km))	30.05	28.59

5.3.4 Combination of several events

We have seen in the previous paragraphs the results for isolated events, such as only one roundabout or one toll. In order to estimate the benefit of a combination of several driving events, we create the digital signal of the road where we carried out our data collection. The 125km-road is presented in Section 4.2.3 and Fig. 4.6 and contains a lot of driving events from urban, extra-urban roads and highways.

We perform the whole process previously illustrated in Fig. 5.1. Therefore, we run the self-learning cycle in simulation, and we use our Matlab tool to adapt the initial cycle to an efficient driving.

The simulation results are presented in Fig. 5.11 and Table 5.6. The differences between drivers are expressed by 52 coasting phases (roundabouts, tolls, and stops) and 5 stop times. These changes represent 9.9% of the relevant events (*i.e.*, events with a delta distance between them greater than 10 meters), and 5.8% of the total number of speed changes of this cycle. These changes imply 1.5% of fuel savings and an average speed 3% greater for an efficient driver than for a non-efficient one.

These scores will be higher and increasingly relevant when other driving behavior differences on events are added. However, the fuel consumption and average speed values observed for only a few events are sufficient enough to start implementing different driving behaviors in terms of rational driving.

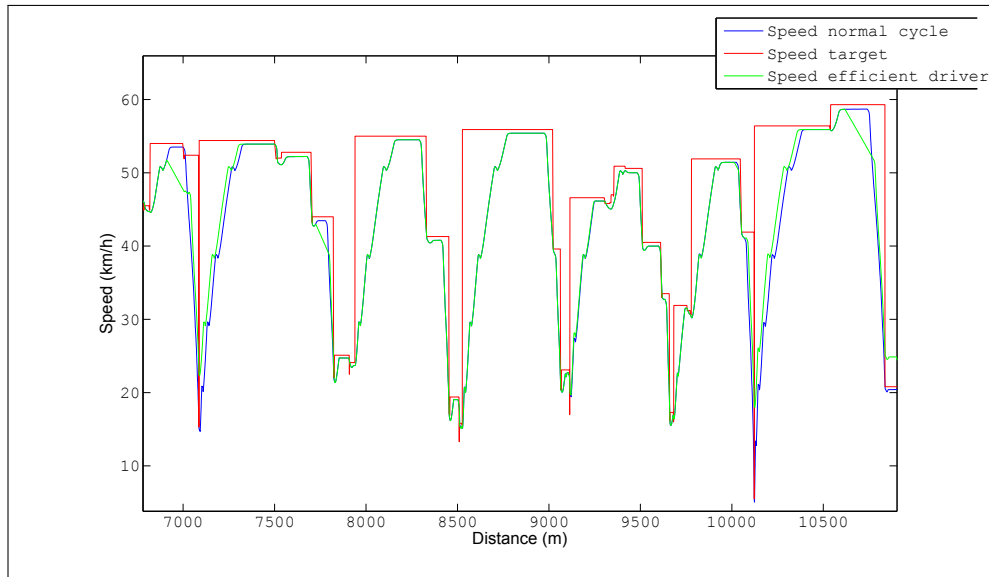


Figure 5.11: Speed signals for an efficient (green) and a non-efficient (blue) driver on the same cycle (red). There are several phases of coasting (7100m, 7800m, 10600m). To ensure readability, only a few km are represented here and not the 125km-long cycle.

Table 5.6: Average speed, f_c , and RDI results for the reference cycle (non-efficient) and the efficient driving cycle.

Driver type	Reference = Non efficient	Efficient
RDI	1.91	1.96
Average speed (km/h)	53.5	53.96
Average f_c (L/(100km))	28.01	27.6

Moreover, we can measure if our model provides speed curves closer to real driving than the previous model. To this end, we perform a simulation from a cycle corresponding to a real efficient driver.⁴ We measure the distance between the speed

⁴Creating a cycle from real data is made with an internal tool, referred as cycle generator in Fig. 5.1.

curves of the efficient real driver and the two simulations, namely efficient and non-efficient. This distance is measured with the Euclidean norm⁵ of the difference of the two speed signals. A zoom on a roundabout approach can be seen on Fig. 5.12. The distance between the real driving speed and the simulated efficient speed is 700.5 (km/h), whereas the distance between the real driving speed and the simulated non-efficient speed is 716.24 (km/h). Therefore, the new driver model implemented during this work gives closer speed curves to real ones than the former driver model.

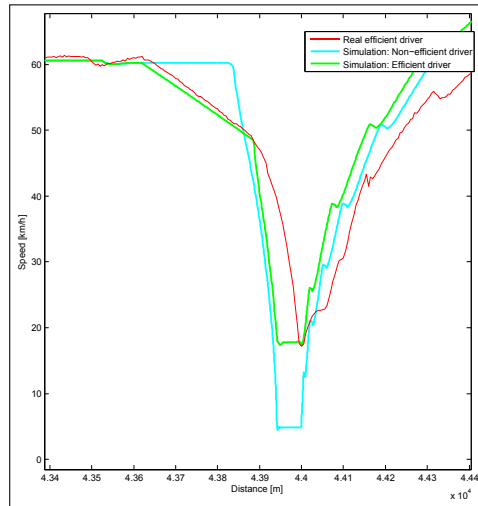


Figure 5.12: Zoom on a roundabout approach: speed signals for an efficient (green) and a non-efficient (blue) driver on the same cycle from an efficient driver (red).

5.3.5 Summary

In this chapter, we presented and detailed the structure of the generator of different driver behaviors. The used method is to adapt a reference cycle to the desired driver and to the target vehicle dynamics. To this end, we apply the results obtained in Chapter 3 concerning the differences in driver behaviors in terms of rational driving, and we learn the acceleration profiles of the vehicle with a self-learning cycle. The cycle preprocessed with this information can be used in the usual simulation and chassis-dynamometer control systems.

The implementation of driving differences in terms of rational driving was presented for roundabout, toll and stops, and provided relevant results in terms of fuel consumption, average speed, and RDI. We also tested the implementation on a full cycle containing many urban, extra-urban and highway driving situations. The tests on chassis-dynamometer are ongoing and will be the subject of an internal

⁵On an n -dimensional Euclidean space R^n , the Euclidean norm of the vector $x = (x_1, x_2, \dots, x_n)$ is: $\|x\| = \sqrt{x_1^2 + \dots + x_n^2}$

technical report. On simulation, we obtain differences between drivers in terms of fuel consumption and average speed by only changing roundabouts, tolls and stops, which represent 9.9% of our test cycle (since the others event are not covered so far). Moreover, the speed curves are closer to a real driving than the former driver model. However, analyzing all the other driving events will be necessary to cover the widest possible range of driving situations and to be even closer to real driving.

Conclusion

In a context of global fuel saving awareness and cost reduction, virtual testing during the truck development process is a necessity. Having a realistic driver model in terms of actuator control and speed is a benefit to accurately estimate fuel consumption and average speed of the future trucks tested. We have approached driver behavior analysis using real driving data to create different driver behaviors regarding fuel consumption and average speed issues. To this end, we have proposed a two-step method: first, the identification of driving events through driving features such as maximum speed. Second, the quantification of three different driving behaviors on the recognized driving events. Finally, the purpose was to implement our results in a tool that creates different driving behaviors. The output of this tool is a cycle adapted to a driver type, and this cycle can be used in the two targeted test facilities, namely the simulation tool and the chassis-dynamometer.

Using collected data, we proposed an automatic driving event recognition method with the selection of relevant driving features and adapted it to simulation data. We then presented the concept of rational driving and we proposed a Rational Driving Index to rank the drivers depending on their ability to drive rationally. Then we estimated the correlation of some driving features concerning driving behaviors, *e.g.* the crossing speed in roundabouts, or the use of coasting, with this indicator. We also compared the RDI to a cost model and proved its relevancy. Once the driving features that impacted RDI had been analyzed, we proposed a comprehensive structure to adapt a selected road to a desired driver type and vehicle, relatively to a reference driver type. The output of this structure is directly usable in our usual tools to obtain reliable vehicle performance data, *i.e.*, fuel consumption and average speed, adapted to the desired driver type.

The classification of driving events reached a classification rate of 74% and 91% for $Throttle = 0$ events and $Throttle > 0$ events, respectively. On simulated data the accuracies range from 80% to 90%. These results can seem low for $Throttle = 0$ events, but they are satisfactory for our purpose due to the raw nature of driving events and the similarities between the different classes. Features selection proves that minimum speed, maximum speed, delta speed, road type, and stop are the most differentiating parameters for driving events recognition. A direct comparison with the results of related works is not feasible because both the driving data and the driving events to be recognized, considered in this work, are different from those found in the literature. However, instead of the two learning algorithms chosen,

we also tested the instance-based learning k-NN method used in [6] and [7]. We conducted a series of experiments varying k , and we found that, where $k = 4$, k-NN achieves the best classification rate, namely 73% for $Throttle=0$ events and 91% for $Throttle>0$ events with a false positive rate of 6.2% and 4.5%, respectively. These results prove to be close to the accuracies achieved by the two learning algorithms, and remain comparable to the performance observed in the literature, even though the compared works are not identical as both event types and chosen driving data are different.

We obtained promising results concerning the correlation between driving features and RDI. We focused especially on typical events, namely roundabout on extra-urban roads, toll on highways, and stop on urban roads. We estimated the correlation between coasting and RDI on real and on simulated data for toll and roundabout approach. Concerning roundabout, we have also found that efficient drivers tend to cross the roundabouts at higher speeds than other drivers. To our knowledge, the correlation of driving features, computed over the length of driving events that only last a few seconds, with a rational driving index for the purpose of fuel consumption and average speed analysis has not been done before. Thus, comparing our results is also difficult for this part of our work. We performed the correlations with the usual features that are proven to be fuel consumption-related features but in related works the features were not adapted to a truck context, or not measured by event but rather over longer time scales.

We chose to create different driving behaviors by preprocessing the road cycle, as can be done in other driver models. We modified some driving parameters (*e.g.*, speed target), but we also added the actuator codes. They are important since they prevail over the target speed. Even if the choice of target speed is specific to a driver style, *e.g.*, an efficient driver will have high roundabout crossing speed, a driver controls the vehicle primarily by using the actuators. Thus, creating setpoints for actuator utilization seems to be relevant and really adapted to our work.

The results of this implementation prove to be efficient since we simulated different driving behaviors, and obtained interesting fuel consumption and average speed values on the same road cycle but adapted to different driver behaviors. On a full approach, adapting only 10% of the events of a cycle produces fuel saving of around 1.5% and an average speed which is 3% faster for an efficient driver.

Despite these results, we encountered some limitations during our work. The first one is the number of driving events implemented. Whereas the recognized events cover as many longitudinal driving events as possible, not all these events were implemented on our Cycle Tuning tool. We built a table containing all the driving events and their priority for implementation with our internal expert drivers. We then implemented the events by order of priority. In future work, the number of events impacting differences in terms of fuel consumption and average speed should be more higher. It will be particularly important in the short term to add driving

behavior differences for take-offs and higher speed limit events. While we dealt with the main *throttle* = 0 events in this work, the *throttle* > 0 events should be the next priority as the throttle position is directly linked to fuel injection. However, we chose to focus on some *throttle* = 0 events first for the purpose of energy saving recovery where coasting and braking are highly important. Moreover, although we developed methods to differentiate driving behaviors in terms of rational driving, another prospect is to find the theoretical upper limit of the best possible rational driving.

Moreover, the number of driver types could be also increased, but moderately since having a wide diversity of behaviors also increases the testing time. Three classes is a good compromise, but we could differentiate driver types more accurately, *i.e.*, weighted by speed, such as *efficient but slow driver*, or *efficient and fast driver*. These distinctions can be useful when testing ADASs. For example, an ADAS will not have the same positive impact on a *fast and non-efficient* driver as on a *slow and non-efficient* one, though they are both non-efficient.

Another perspective that has emerged is the analysis of the chain of events. We have seen that efficient drivers tend to use the kinetic energy of their vehicle more efficiently and consequently they brake and re-accelerate less often than other drivers. This impacts the number directly the number of events, and also the sequel of events. Therefore the analysis of the chain of events could reveal driving behavior differences between drivers and could be implemented by adapting the number and the kind of events of the reference cycle.

Truck usages

Segmentation of trucks usage from GTA are presented in Table A.1 (Volvo group internal classification) and ACEA segmentation is presented in Table A.2 (European Manufacturers classification).

Table A.1: Truck usages from GTA





Operating cycle	Description	Characteristics	Truck example
Long Distance	The mean distance between delivery or pick-up of goods/passengers is longer than 50km	Few stops, high average speed, long distances covered during each working period	
Regional	The mean distance between delivery or pickup of goods/passengers is shorter than 50 km but longer than 5km	Mixture of short and long distances between stops, stops are usually clustered	
Local	The mean distance between delivery or pickup of goods/passengers is shorter than 5 km but longer than 0.5 km	Many stops, low average speed	
Stop and Go	The mean distance between delivery or pickup of goods/passengers is shorter than 0.5 km	Stop and go, low speed, stationary	

Table A.2: Truck usages ACEA

Vehicle cycle	Description	Average yearly run distance (km)
Long haul	Delivery to national and international sites (mainly highway operation and a small share of regional roads)	135.000
Regional delivery	Regional delivery of consumer goods from a central store to selling points (innercity and partly suburban roads)	60.000
Urban delivery	Urban truck operation like refuse collection (many stops, partly low vehicle speedoperation, driving to and back to central base point)	40.000
Municipal utility	Urban delivery of consumer goods from a central warehouse to local sotres (innercity, suburban, regional and also moutain roads)	25.000
Construction	Construction site vehicle with delivery from central sotre to very few local customers (innercity, suburban and regional roads; only small share of off-road driving)	60.000

List of driving events

This appendix presents an as comprehensive as possible list of all driving events. This list is built with the help of our internal expert drivers. We also provide an estimation of actuators used during each event, the sensibility to fuel savings to driver type, the sensibility to vehicle, the relief impact and the road type.

		A = THROTTLE		1		High		++		Highway = at least two lanes in each direction, separated by a median		
		B = COASTING		2		Medium		+		Main road = 1 lane in each direction, speed limit higher than 50km/h		
		C = RETARDER(S)		3		Low		0		Urban = Speed limit lower or equal to 50km/h		
		D = SERVICE BRAKES										
Event class	Environment event	Variant	A / B / C / D States	Driver sensibility to fuel economy	Vehicle dynamic sensibility	Relief impact	H / M / U Road type					
Mandatory stop	Toll (on motorway)		x x x x	High	3	+	H					
			x x x x	Medium	2	+						
			x x x x	Low	1	0						
	Stop Sign		x x x x	High	3	++	M / U					
			x x x x	Medium	2	++						
			x x x x	Low	1	0						
	Frozen traffic jam		x x x x	High	3	++	H / M / U					
			x x x x	Medium	2	+						
	Zero anticipation (e.g. short distance from traffic light switching to red)			x	All	1	0	H / M / U				
Urban delivery			x x	All	2	+	U					
Logistic platform delivery			x x	All	2	+	M / U					
Driving time (mandatory breaks), motorway							H					
Driving time (mandatory breaks), others			x x x	All	3	+	M / U					
Unsystematic stop	Give way (including roundabouts entry)	No Stop	x x	High	3	++	M / U					
		Stop	x x x	Medium	2	+						
		No Stop	x x x	Low	2	0						
		Stop	x x x	High	3	++						
		No Stop	x x	Medium	2	+						
		Stop	x x	Low	1	+						
	Traffic lights Hypothesis: always coming with a 50km/h speed in front	Green	x		High	3	++	U				
		Red	x x x		Medium	2	+					
		Green	x		Low	1	+					
		Red	x x x		High	3	++					
		Green	x		Medium	2	+					
		Red	x x x		Low	1	+					
	Priority to the right (in town, w/o X sign)	No Stop	x x x		Medium	2	+	U				
		Stop	x x x		Low	1	0					
		No Stop	x x		High	3	+					
		Stop	x x		Medium	2	0					
	Priority to the right (out of town, with X sign)	No Stop	x x		High	3	+	M				
		Stop	x x x		Medium	2	0					
		No Stop	x x		High	3	+					
		Stop	x x x		Medium	2	0					
Pedestrian crossing	No Stop	x		High	3	+	U					
	Stop	x x x		Medium	2	0						
	No Stop	x		High	3	+						
	Stop	x x x		Medium	2	0						
Traffic jam (accordon)			x x	High	1	0	H / M / U					
			x x	Low	1	0						
Speed augmentation required	Take-off from standstill with target speed (neither visibility nor information on next speed limit)	Up to 80-90 km/h	x	High	3	++	H / M					
		Up to 50 km/h	x	High	3	++						
		Up to 80-90 km/h	x	Medium	2	+						
		Up to 50 km/h	x	Medium	2	+						
		Whatever target	x	Low	1	0						
		Whatever target	x	Low	1	0						
	Take-off from standstill with target speed with visibility or information on next (and lower) speed limit		x		High	3	++	H / M / U				
			x		Medium	2	+					
	In motion (overtaking becoming possible)		x		All	1	0	M / U				
			x		All	1	0					
	In motion, either traction or retardation (new higher speed limit)	Throttle > 0	x		High	3	++	H / M / U				
		Throttle = 0	x x		High	3	++					
Throttle > 0		x		Medium	2	+						
Throttle = 0		x x		Medium	2	+						
Whatever the slope		x		Low	1	0	H / M / U					
		x		Low	1	0						
Speed reduction required (no stop)	Speed bumps, chicanes...		Same as give way / no stop									
	Lower limit from stabilized speed		x x x	High	3	++	H / M / U					
			x x x	Medium	3	+						
			x x x	Low	1	0						
	Lower limit from increasing speed (e.g. current target = 70km/h, but next is 50km/h at short distance)		x x	High	3	++	M / U					
			x x x	Medium	3	+						
			x x x	Low	1	0						
	Curves w/o specific speed limitation (see above if associated to a speed limitation)		x x x	High	3	++	H / M					
			x x x x	Medium	3	+						
			x x x	Low	3	0						
	Road state		Number of events is not significant (construction not considered)									
	Weather conditions		Main difference is b/w dry / wet roads									
	Clearance limitations (width & height) w/o specific speed limitation		Each driver sets his own speed limitation									
	Slower vehicle (w/o possibility of immediate overtake)		See lower speed limit									
Downhill w/o specific speed limitation		x x x	High	3	++	H / M						
		x x x	Medium	2	++							
		x x x	Low	1	+							
Voluntary Speed reduction	Due to relief and driver education (top of a hill)		x	High	3	+++	H / M					
			x	Medium	2	++						
			x	Low	1	0						
Stabilization of the speed	Downhill (pente assez élevée pour ne pas avoir besoin de maintenir l'accélérateur)		x x x	High	3	++	H / M					
			x x x	Low	1	0						
Stabilized speed	Stabilize speed w/ stable resistive efforts		x	All	1	0	H / M / U					
	Stabilize speed w/ increasing resistive efforts (until engine torque saturation)		x	High	3	+++		H / M				
			x	Low	1	+						
	Stabilize speed w/ decreasing resistive efforts (until coasting)		x x	High	3	+++		H / M				
		x x	Low	1	+							

Figure B.1: Table of driving events.

Driving features in related work

In the following tables:

v is the vehicle velocity,

a is the vehicle acceleration,

a^+ is the positive acceleration,

x is the total distance,

dv is the delta velocity,

dt is the delta time,

d_r is the relative distance from the preceding vehicle,

v_f is the final vehicle speed on the corresponding section,

v_i is the initial vehicle speed on the corresponding section,

v_r is the relative speed with respect to the preceding vehicle,

X or x indexes correspond to the signal value on the longitudinal axis,

Y or y indexes correspond to the signal value on the lateral axis.

S.D. means standard deviation.

Table C.1: Driving features

Large scale driving patterns		
Reference	Goal	Driving features
[8]	Effect on fuel utilization and exhaust emission factors	<p>Average speed, S.D. of speed, average acceleration, S.D. of acceleration, average deceleration, deceleration S.D., number of acceleration/deceleration shifts per 100m where the difference between adjacent local max-speed and min-speed was $> 2km/h$, number of acceleration/deceleration shifts per 100m where the difference between adjacent local max-speed and min-speed was $> 10km/h$, number of acceleration/deceleration shifts per 100s where the difference between adjacent local max-speed and min-speed was $> 2km/h$, number of acceleration/deceleration shifts per 100s where the difference between adjacent local max-speed and min-speed was $> 10km/h$, relative positive acceleration ($RPA = \frac{1}{x} * \int (v * (a^+)) dt$), integral of the square of the acceleration, % of time when speed $< 2km/h$, average stop duration, number of stops per kilometer, % of time in speed interval 0 – 15km/h, % of time in speed interval 15 – 30km/h, % of time in speed interval 30 – 50km/h, % of time in speed interval 50 – 70km/h, % of time in speed interval 70 – 90km/h, % of time in speed interval 90 – 110km/h, % of time in speed $> 110km/h$, % of time in deceleration interval $-10; -2.5m/s^2$, % of time in deceleration interval $-2.5; -1.5m/s^2$, % of time in deceleration interval $-1.5; -1m/s^2$, % of time in deceleration interval $-1; -0.5m/s^2$, % of time in deceleration interval $-0.5; 0m/s^2$, % of time in acceleration interval $0; 0.5m/s^2$, % of time in acceleration interval $0.5; 1m/s^2$, % of time in acceleration interval $1; 1.5m/s^2$, % of time in acceleration interval $1.5; 2.5m/s^2$,</p>

Reference	Goal	Driving features
[8] (suite)	Effect on fuel utilization and exhaust emission factors	% of time in acceleration interval $2.5; 10m/s^2$, average engine speed, engine speed S.D., % of time with engine speed $0 - 1500rpm$, % of time with engine speed $1500 - 2500rpm$, % of time with engine speed $2500 - 3500rpm$, % of time with engine speed $> 3500rpm$, % of time in gear 3 with engine speed $0 - 1500rpm$, % of time in gear 3 with engine speed $1500 - 2500rpm$, % of time in gear 3 with engine speed $2500 - 3500rpm$, % of time in gear 3 with engine speed $> 3500rpm$, % of time in gear 4 with engine speed $0 - 1500rpm$, % of time in gear 4 with engine speed $1500 - 2500rpm$, % of time in gear 4 with engine speed $2500 - 3500rpm$, % of time in gear 4 with engine speed $> 3500rpm$, % of time in gear 5 with engine speed $0 - 1500rpm$, % of time in gear 5 with engine speed $1500 - 2500rpm$, % of time in gear 5 with engine speed $2500 - 3500rpm$, % of time in gear 5 with engine speed $> 3500rpm$, positive kinetic energy ($PKE = (\sum(v_f^2 - v_i^2))/x$ when $\frac{dv}{dt} > 0$, % of time when $va < 0m/s^3$, % of time when va is $0 - 3m/s^3$, % of time when va is $3 - 6m/s^3$, % of time when va is $6 - 10m/s^3$, % of time when va is $10 - 15m/s^3$, % of time when $va > 15m/s^3$, average va .

Reference	Goal	Driving features
[35]	Determination of the most effective driving features for hybrid electric vehicle (HEV) intelligent control	Mean of velocity, variance of velocity, energy, variance of energy, idle time percentage, maximum velocity, minimum velocity, $max_{velocity} - min_{velocity}$, mean of acceleration, variance of acceleration, AA = average accelerating, AD = average decelerating, RPA = relative positive acceleration, RNA = relative negative acceleration, cruise percentage, max acceleration, min acceleration displacement, mean square acceleration.
[34]	Examine the effect on driving behavior of different driver categories and the characteristics of the local environment in which the car is driven	Average deceleration, RPA , % of time speed is $< 2km/h$, frequency of local max/min values of the speed curve per 100s, % of time when va is $3 - 6m/s^3$, % of time at acceleration over $1.5m/s^2$, % of time at speeds of $15 - 30km/h$, % of time at speeds of $90 - 110km/h$, % of time at speeds of $70 - 90km/h$, % of time at speeds of $50 - 70km/h$, % of time engine speed $2500 - 3500rpm$ in 3^{rd} gear, % of time engine speed is $> 3500rpm$, % of time speed is $> 110km/h$, % of time engine speed $1500 - 2500rpm$ in 2^{nd} gear, % of time engine speed is $< 1500rpm$ in 4^{th} gear, % of time engine speed is $< 1500rpm$ in 5^{th} gear.
[17]	Recognition of drivers	Histograms of brake and gas pedals \rightarrow (Dynamic features histograms) \rightarrow GMM parameters of the histograms, Cepstrum of brake and gas pedals \rightarrow FFT to keep lower order coefficients \rightarrow (Dynamic features histograms) \rightarrow GMM parameters of the histograms.

Reference	Goal	Driving features
[16]	Road type recognition	Means and standard deviations of the normalized following data: Acceleration pedal position, Gear, Turn indicator flash, Vehicle speed, Steering angle, Steering angle change rate, Steering angle sign, Brake pedal active.
Small-scale driving patterns		
Reference	Goal	Driving features
[6]	Recognize aggressive driving through driving events	Gyroscope values: $G = \{g_x; g_y; g_z\}$ in rad/s Device accelerometer values $A = \{a_x; a_y; a_z\}$, Euler angle rotation $E = \{e_x; e_y; e_z\}$ in rad , Features vector: $T = \{g_x; a_y; e_x\}$ and $L = \{g_y, a_z\}$. Features are templates obtained with DTW algorithm for each recorded event, with A , G and T sets of signals. The DTW algorithm is designed to find an optimal alignment of two signal vectors. In their case, they aligned the currently detected event signal with the pre-recorded template signals.
[7]	Recognize driver's style	Mean values and standard deviations of the following variables: On <i>go</i> phases: slope, max slope. On <i>track</i> phases: headway time ($T_H = \frac{d_r}{v}$) and its S.D. On <i>stop</i> phase: slope, max slope, v_r and T_H . On <i>cruise</i> phase: not considered
[11]	Differentiate two drivers through acceleration, braking and/or turning events	Acc. Feature Vector and Brake Feature Vector: $\{Histogram(AccY; 5), Min(AccY), Max(AccY), Mean(AccY), Variance(AccY), Duration(AccY)\}$, Turning Feature Vector: $\{Histogram(Gyro; 5), Min(GyroSignal), Max(Gyro), Mean(Gyro), Variance(Gyro), Histogram(AccY; 5), Min(AccY), Max(AccY), Mean(AccY), Variance(AccY), Duration(AccY)\}$.
[9]	Multimodal study into the possibility of using naturalistic driving studies (NDS) for automated drive analysis	Lane features: lateral deviation of the car from center of the lane, yaw angle of the car, vehicle speed and width of the lane. Maps, <i>i.e.</i> , information about the road from the digital map database: type of road, speed limits and warnings that are seen at different points during the drive. CAN data: Speed.

Reference	Goal	Driving features
[19]	Powered two-wheelers riding patterns recognition	Raw accelerometers and gyroscopes measurements.
[10]	Traffic flow analysis	Number of deceleration events, speed of the vehicles, number of vehicles, number of lane changes.
[3]	Rational driving analysis	Anticipation of driving events, use of the kinetic energy of the vehicle, use of the right engine speed, crossing speed on curves and roundabouts.
[25]	Predictability of drivers	S.D. of Longitudinal Acceleration, S.D. of Longitudinal Jerk, S.D. of Lateral Acceleration, S.D. of Lateral Jerk, Average Time-gap to Lead Vehicle.
[21]	Driving events recognition	Segmentation of the following data: Longitudinal acceleration, Lateral acceleration, horizontal rotation, road slope, velocity.
[22]	Score driving behaviors	Threshold on longitudinal and lateral acceleration
[23]	Estimate driver behavior (safe or aggressive)	Templates describing speed, acceleration, deceleration, deflection angle
[24]	Evaluate driving behavior mobile application	Threshold on longitudinal and lateral acceleration.
[20]	Estimate driver behavior (calm or aggressive)	RPM, Speed, Absolute Throttle Position, Calculated Load, Fuel Rate, Boost Pressure, Instant Economy, O2 Sensor B1S2, O2 Sensor B1S1, Short Fuel Trim B1S2, Ignition Timing Advance, Air Flow Rate for MAP, Short Term Fuel Trim B1, Fuel Cost

Table C.2: Large scale driving patterns recognition methods and results

Reference	Goal	Methods and Results
Large scale driving patterns		
[8]	Effect on fuel utilization and exhaust emission factors	
[35]	Determination of the most effective driving features for hybrid electric vehicle (HEV) intelligent control	
[34]	Examine the effect on driving behavior of different driver categories and the characteristics of the local environment in which the car is driven	
[17]	Recognition of drivers	Log-likelihood. Accuracies between 47.5% and 89.6% depending on the database (simulator or real, number of drivers and length of signals (3 or 5 minutes)). With real data only: from 47.5% to %76.8.
[16]	Road type recognition	For supervised training, a standard multi-layer perceptron (MLP) with 8 inputs (one for each input variable), 20 hidden and 4 output units was used. For unsupervised learning, a standard competitive learning network with eight inputs and four outputs was used. Results 93.06% on data test.

Table C.3: Small-scale driving events recognition methods and results

Reference	Goal	Events	Methods and Results
Small-scale driving patterns			
[6]	Recognize aggressive driving through driving events	Right turns, Left turns, U-turns, Aggressive right turns, Aggressive left turns, Aggressive U-turns, Aggressive acceleration, Aggressive braking, Swerve right (aggressive lane change), Serve left (aggressive lane change), Device removal, Excessive speed.	kNN with $k = 3$. Accuracies from 77% to 91%, with a false positive rate of 5%.
[7]	Recognize driver's style		k-NN classify new driver belonging to each cluster (normal, quiet, aggressive, for each phase). No accuracies.
[11]	Differentiate two drivers through acceleration, braking and/or turning events	Brake event ($brake\ light = 1$), acceleration event ($acceleration\ pedal > 1$), turning event ($ steering\ angle > 30^\circ$).	SVM and k-mean clustering ($k = 2$). Accuracies from 0.54 to 0.85

Reference	Goal	Events	Methods and Results
[9]	Multimodal study into the possibility of using naturalistic driving studies (NDS) for automated drive analysis	Drive analysis characteristics: Number of right lane changes, Number of left lane changes, Time spent on freeway, Time spent on urban road, Total distance, Average speed on freeway, Average speed on urban road, Number of stops, Number of right turns, Number of left turns, Number of freeway entries, Number of freeway exits, Time spent on single lane, Number of right curves, Number of left curves, Average distance from center of lane.	
[19]	Powered two-wheelers riding patterns recognition	Left Turn, Right Turn, Straight Line, Roundabout, Stop.	Gaussian mixture models (GMM), k-Nearest Neighbors (k-NN), Support Vector Machines (SVM), Random Forests (RF) and discrete and continuous Hidden Markov Models (HMM). Rates from 60% to 96% depending on the method and the driver
[10]	Traffic flow analysis		Correlation between features and time when traffic jam occurred
[3]	Rational driving analysis		
[25]	Predictability of drivers		

Reference	Goal	Events	Methods and Results
[21]	Driving events recognition	Left and right curves, Left and right turns, Turning a vehicle left and right with on roundabouts, Driving straight across an intersection with a roundabout.	HMMs. 98.3% on 238 manually labeled events
[22]	Score driving behaviors	Lane drifting, lane warning, acceleration, braking, turning	No classification.
[23]	Estimate driver behavior (safe or aggressive)	Sharp maneuvers, unsafe right and left turns, lane departures, sudden braking, sudden speed-up.	Match training safe template with DTW. Then Bayesian inference: maximum a posteriori estimates across different events. Event type and its safe/unsafe driving style is found correct at 93.3%.
[24]	Evaluate driving behavior mobile application	Acceleration, Braking, Turn left, Turn right.	No classification.
[20]	Estimate driver behavior (calm or aggressive)		

Data collection documents

D.1 Driver questionnaire

Questionnaire suivi des essais

Merci de remplir le questionnaire suivant qui a pour but de nous permettre de faire un suivi de nos essais. Ces données seront strictement **confidentielles** et traitées de façon **anonyme**. Elles ne seront **jamais** communiquées en dehors de notre service.

Nom :

Prénom :

Age :

Sexe : H F

Années d'obtention des permis (C, D, E) :

Années d'expérience du métier de chauffeur routier :

Historique des entreprises de transport dans lesquelles vous avez travaillé (les plus récentes) :

Généralement vous avez travaillé sur des trajets : (plusieurs réponses possibles)

- Urbain
- Régional
- National
- International

Votre kilométrage annuel ces dernières années (poids-lourd seulement) :

- <30000 km
- 30000-60000 km
- 60000-90000 km
- 90000-120000 km
- >120000 km

Types de véhicule conduits (les plus récents uniquement) : (plusieurs réponses possibles)

- Tracteur
- Porteur
- Bus
- Car

Pour les porteurs, configuration d'essieux, de PTAC et de carrosserie :

Est-elle facile d'utilisation ?

- Oui
- Non

Utilisez-vous le couplage des ralentisseurs à la pédale de frein ?

- Oui
- Non

Utilisez-vous les régimes moteurs élevés en phase de ralentissement ?

- Oui
- Non

Vos remarques sur les ralentisseurs lors de l'essai :

L'essai :

Trafic du roulage de l'après-midi :

- Fluide
- Ralentie (si oui, à quel endroit ?)
- Bouchons (si oui, à quel endroit ?)

Avez-vous des remarques sur les essais faits aujourd'hui (véhicule, parcours etc.) :

D.2 Accompanying person questionnaire

Questionnaire suivi des essais

Accompagnateur :

Date de l'essai :






Nom du chauffeur :

Heure de départ (matin) :

Heure d'arrivée (matin) :

Heure de départ (après-midi) :

Heure d'arrivée (après-midi) :

Météo :       

Température extérieure :

Vent :

- Nul
- Modéré
- Fort
- Pas d'avis

Trafic :

- Fluide
- Ralenti (si oui, à quel endroit ?)
- Bouchons (si oui, à quel endroit ?)

Clim / chauffage / fenêtres ouvertes :

Caractériser le chauffeur : (plusieurs réponses possibles)

- Conduite rationnelle
- Conduite normale
- Conduite binaire (accélération/coups de frein)
- Nerveux
- Lent
- Rapide
- Concentré
- Distrait
- Mou
- Respectueux des limites de vitesses
- Dangereux (dans les courbes, les ronds-points...)
- Autre (précisez)

Remarques sur le chauffeur :

Autres remarques :

Bibliography

- [1] T. Bellet and H. Tattegrain-Veste. Cosmodrive: un modèle de simulation cognitive du conducteur automobile. *Spérando*, JC, Wolf, M.,(sous la direction de), *Formalismes de modélisation pour l'analyse du travail et l'ergonomie*. Presse Universitaire de France, Paris, pages 77–110, 2003. (Quoted on pages ix, 45 and 46.)
- [2] T. Bellet, B. Bailly-Asuni, P. Mayenobe, and A. Banet. A theoretical and methodological framework for studying and modeling drivers' mental representations. *Safety Science*, 47(9):1205–1221, 2009. (Quoted on pages ix, 45 and 47.)
- [3] A. Maincent. *Comportements humains, Activités finalisées et Conception de systèmes d'assistance à la conduite de Véhicules Industriels*. PhD thesis, Université Lumière Lyon II, 2010. (Quoted on pages 4, 5, 10, 25, 27, 74, 78, 85, 87, 134 and 137.)
- [4] ADEME. Rouler plus responsable. <http://ecocitoyens.ademe.fr/mes-deplacements/adopter-leco-conduite/rouler-plus-responsable>, 2013. (Quoted on page 4.)
- [5] S. Fornengo. Projet "conduite économique assistée". *Rapport intermédiaire d'avancement. Convention ADEME n 03 66 027*, 2004. (Quoted on page 5.)
- [6] Derick A Johnson and Mohan M Trivedi. Driving style recognition using a smartphone as a sensor platform. In *14th International IEEE Conference on Intelligent Transportation Systems (ITSC), 2011*, pages 1609–1615. IEEE, 2011. (Quoted on pages 10, 12, 20, 21, 23, 25, 26, 28, 29, 67, 122, 133 and 136.)
- [7] M. Canale and S. Malan. Analysis and classification of human driving behaviour in an urban environment. *Cognition, Technology & Work*, 4(3):197–206, 2002. (Quoted on pages 10, 20, 23, 26, 28, 30, 67, 122, 133 and 136.)
- [8] Eva Ericsson. Independent driving pattern factors and their influence on fuel-use and exhaust emission factors. *Transportation Research Part D: Transport and Environment*, 6(5):325–345, 2001. (Quoted on pages 10, 26, 27, 28, 62, 130, 131 and 135.)
- [9] R.K. Satzoda, S. Martin, M. Van Ly, P. Gunaratne, and M. M. Trivedi. Towards automated drive analysis: a multimodal synergistic approach. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC), 2013*. IEEE, 2013. (Quoted on pages 12, 20, 21, 22, 26, 28, 133 and 137.)
- [10] M. Litzenberger, H. Glasl, B. Kohn, B. Schalko, and G. Fernandez. Sensor fusion on an embedded system for traffic data analysis - etrada-v system. In

- Proceedings of the 11th International IEEE Conference on Intelligent Transportation Systems Beijing, China*, 2008. (Quoted on pages 12, 20, 21, 22, 28, 134 and 137.)
- [11] M. Van Ly, S. Martin, and M. M. Trivedi. Driver classification and driver style recognition using inertial sensors. In *IEEE Intelligent Vehicles Symposium*. IEEE, 2013. (Quoted on pages 12, 20, 21, 22, 26, 28, 30, 133 and 136.)
- [12] Paolo Giudici and Silvia Figini. *Applied data mining for business and industry*. John Wiley & Sons, 2009. (Quoted on page 17.)
- [13] G. Piatetsky-Shapiro. Knowledge discovery in real databases: A report on the ijcai-89 workshop. *AI Magazine*, 11(5):68–70, 1991. (Quoted on page 18.)
- [14] Usama M. Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. Advances in knowledge discovery and data mining. chapter From Data Mining to Knowledge Discovery: An Overview, pages 1–34. American Association for Artificial Intelligence, Menlo Park, CA, USA, 1996. (Quoted on page 18.)
- [15] Ronald J. Brachman and Tej Anand. Advances in knowledge discovery and data mining. chapter The Process of Knowledge Discovery in Databases, pages 37–57. American Association for Artificial Intelligence, Menlo Park, CA, USA, 1996. (Quoted on page 18.)
- [16] J. Engström. Real-time recognition of driver behaviour, dissertation for master of science degree evolutionary and adaptive systems, 2001. (Quoted on pages 19, 25, 28, 133 and 135.)
- [17] Chiyomi Miyajima, Yoshihiro Nishiwaki, Koji Ozawa, Toshihiro Wakita, Katsunobu Itou, Kazuya Takeda, and Fumitada Itakura. Driver modeling based on driving behavior and its evaluation in driver identification. *Proceedings of the IEEE*, 95(2):427–437, 2007. (Quoted on pages 19, 20, 23, 25, 28, 30, 132 and 135.)
- [18] M. Platho, H-M. Groß, and J. Eggert. Learning driving situations and behavior models from data. In *16th International IEEE Annual Conference on Intelligent Transportation Systems*, 2013. (Quoted on pages 20 and 25.)
- [19] F. Attal, A. Boubezoul, L. Oukhellou, and S. Espie. Riding patterns recognition for powered two-wheelers users behaviors analysis. In *16th International IEEE Annual Conference on Intelligent Transportation Systems*, 2013. (Quoted on pages 20, 21, 26, 28, 30, 134 and 137.)
- [20] O Karaduman, H Eren, H Kurum, and M Celenk. An effective variable selection algorithm for aggressive/calm driving detection via can bus. In *Connected Vehicles and Expo (ICCVE), 2013 International Conference on*, pages 586–591. IEEE, 2013. (Quoted on pages 20, 22, 28, 134 and 138.)

-
- [21] Dejan Mitrovic. Reliable method for driving events recognition. *Intelligent Transportation Systems, IEEE Transactions on*, 6(2):198–205, 2005. (Quoted on pages 21, 28, 29, 134 and 138.)
- [22] Luis M. Bergasa, Daniel Almería, Javier Almazán, J. Javier Yebes, and Roberto Arroyo. Drivesafe: an app for alerting inattentive drivers and scoring driving behaviors. In *IEEE Intelligent Vehicles Symposium*. IEEE, 2014. (Quoted on pages 21, 28, 134 and 138.)
- [23] Haluk Eren, Semiha Makinist, Erhan Akin, and Alper Yilmaz. Estimating driving behavior by a smartphone. In *Intelligent Vehicles Symposium (IV), 2012 IEEE*, pages 234–239. IEEE, 2012. (Quoted on pages 21, 22, 28, 134 and 138.)
- [24] Johannes Paefgen, Flavius Kehr, Yudan Zhai, and Florian Michahelles. Driving behavior analysis with smartphones: insights from a controlled field study. In *Proceedings of the 11th International Conference on mobile and ubiquitous multimedia*, page 36. ACM, 2012. (Quoted on pages 21, 22, 28, 134 and 138.)
- [25] Anup Doshi and Mohan M Trivedi. Examining the impact of driving style on the predictability and responsiveness of the driver: Real-world and simulator analysis. In *Intelligent Vehicles Symposium (IV), 2010 IEEE*, pages 232–237. IEEE, 2010. (Quoted on pages 23, 134 and 137.)
- [26] P. Fancher, R. Ervin, J. Sayer, M. Hagan, S. Bogard, Z. Bareket, M. Mefford, and J. Haugen. Intelligent cruise control field operational test. DOT HS 808 849, 1998. (Quoted on page 24.)
- [27] PS Faucher. A comparison of manual versus automatic control of headway as a function of driver characteristics. In *Intelligent Transportation: Realizing the Future. Abstracts of the Third World Congress on Intelligent Transport Systems*, 1996. (Quoted on page 24.)
- [28] Maya Abou Zeid, Ismail Chabini, Edward K Nam, and Alessandra Cappiello. Probabilistic modeling of acceleration in traffic networks as a function of speed and road type. In *The IEEE 5th International Conference on Intelligent Transportation Systems, 2002. Proceedings.*, pages 472–478. IEEE, 2002. (Quoted on page 24.)
- [29] Delphine Delorme and Bongsob Song. Human driver model for smartahs. In *Research Reports, California Partners for Advanced Transit and Highways (PATH), Institute of Transportation Studies, UC Berkeley*, 2001. (Quoted on page 24.)
- [30] S Zorrofi, S Filizadeh, and Paul Zanetel. A simulation study of the impact of driving patterns and driver behavior on fuel economy of hybrid transit buses. In *IEEE Vehicle Power and Propulsion Conference, 2009. VPPC'09*, pages 572–577. IEEE, 2009. (Quoted on page 25.)

-
- [31] U. Kiencke and L. Nielsen. *Chapter 11: Road and driver models*. Automotive control systems for engine, driveline, and vehicle. Springer, 2005. (Quoted on pages 25 and 42.)
- [32] H. Lin, X. Cui, Q. Yu, and S. Yang. Experimental study on diesel vehicle's fuel consumption feature while coasting on level road. *CSIE, Part I, CCIS 152*, pages 264–270, 2011. (Quoted on page 26.)
- [33] Heikki Liimatainen. Utilization of fuel consumption data in an ecodriving incentive system for heavy-duty vehicle drivers. *IEEE Transactions on Intelligent Transportation Systems*, 12(4):1087–1095, 2011. (Quoted on page 26.)
- [34] Karin Brundell-Freij and Eva Ericsson. Influence of street characteristics, driver category and car performance on urban driving patterns. *Transportation Research Part D: Transport and Environment*, 10(3):213–229, 2005. (Quoted on pages 27, 85, 87, 132 and 135.)
- [35] Akbar Naderpour Morteza Montazeri, Abbas Fotouhi. Driving segment simulation for determination of the most effective driving features for hev intelligent control. *Vehicle System Dynamics*, 50(2):229–246, 2012. (Quoted on pages 27, 132 and 135.)
- [36] RL Stratonovich. Conditional markov processes. *Theory of Probability & Its Applications*, 5(2):156–178, 1960. (Quoted on page 30.)
- [37] Geoffrey J McLachlan and Kaye E Basford. Mixture models. inference and applications to clustering. *Statistics: Textbooks and Monographs, New York: Dekker, 1988*, 1, 1988. (Quoted on page 30.)
- [38] D. Aha and D. Kibler. Instance-based learning algorithms. *Machine Learning*, 6:37–66, 1991. (Quoted on pages 31 and 37.)
- [39] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, pages 281–297. University of California Press, USA, 1967. (Quoted on page 31.)
- [40] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995. (Quoted on page 31.)
- [41] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001. (Quoted on page 31.)
- [42] Christopher M Bishop et al. *Pattern recognition and machine learning*, volume 1. springer New York, 2006. (Quoted on page 32.)
- [43] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009. (Quoted on page 33.)

-
- [44] J.R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, 1993. (Quoted on pages 33 and 35.)
- [45] J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986. (Quoted on page 33.)
- [46] Terry Windeatt and Gholamreza Ardeshir. An empirical comparison of pruning methods for ensemble classifiers. In *Advances in Intelligent Data Analysis*, pages 208–217. Springer, 2001. (Quoted on page 35.)
- [47] Jason R Beck, Maria E Garcia, Mingyu Zhong, Michael Georgiopoulos, and Georgios Anagnostopoulos. A backward adjusting strategy for the c4. 5 decision tree classifier. *AMALTHEA REU SITE*, 2007. (Quoted on page 35.)
- [48] Niels Landwehr, Mark Hall, and Eibe Frank. Logistic model trees. *Machine Learning*, 59(1-2):161–205, 2005. (Quoted on page 36.)
- [49] Marc Sumner, Eibe Frank, and Mark Hall. Speeding up logistic model tree induction. In *Knowledge Discovery in Databases: PKDD 2005*, pages 675–683. Springer, 2005. (Quoted on page 36.)
- [50] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer-Verlag, 2001. (Quoted on page 36.)
- [51] Jerome Friedman, Trevor Hastie, Robert Tibshirani, et al. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The annals of statistics*, 28(2):337–407, 2000. (Quoted on page 36.)
- [52] Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, 13(1):21–27, 1967. (Quoted on page 37.)
- [53] P.E. Hart. The condensed nearest neighbor rule. *Institute of Electrical and Electronics Engineers and Transactions on Information Theory*, 14:515–516, 1968. (Quoted on page 37.)
- [54] G.W. Gates. The reduced nearest neighbor rule. *IEEE Transactions on Information Theory*, pages 431–433, 1972. (Quoted on page 37.)
- [55] Belur V Dasarathy. Nosing around the neighborhood: A new system structure and classification rule for recognition in partially exposed environments. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (1):67–71, 1980. (Quoted on page 37.)
- [56] T. Bellet. *Modélisation et simulation cognitive de l'opérateur humain: une application à la conduite automobile*. PhD thesis, Université Paris V, 1998. (Quoted on pages 39 and 45.)

-
- [57] A.J Mc Knight and B.B. Adams. *Driver education task analysis*. Vol I: Task description. Alexandria, Human Resources Research Organisation, 1970. (Quoted on page 40.)
- [58] T.M. Allen, H. Lunenfeld, and G.J. Alexander. *Driver information needs*. Highway Res. Rec., 366, 1971. (Quoted on page 40.)
- [59] G.J.S. Wilde. *The theory of risk homeostasis: implications for safety and health*. Risk Anal. 2, 1982. (Quoted on page 40.)
- [60] Risto Näätänen and Heikki Summala. *Road-user behaviour and traffic accidents*. 1976. (Quoted on page 40.)
- [61] R. Fuller. *A conceptualization of driver behaviour as threat-avoidance*. Ergonomics 27, 1984. (Quoted on page 40.)
- [62] H. van der Molen and A. Botticher. A hierarchical model for traffic participants. pages 557–567, 1988. (Quoted on page 40.)
- [63] E. Hollnagel, A. Nabo, and I.V. Lau. A systemic model for driver-in-control. In *Proceedings of the Second International Driving Symposium on human factors in driver assessment*, Training and Vehicle Design, pages 86–91, 2004. (Quoted on page 40.)
- [64] Jens Rasmussen. Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models. *IEEE Transactions on Systems, Man and Cybernetics*, 13:257–267, 1983. (Quoted on page 40.)
- [65] J. Rasmussen. Information processing and human-machine interaction: an approach to cognitive engineering. Amsterdam, North Holland, 1986. (Quoted on page 40.)
- [66] J.A. Michon. A critical view of driver behavior models: what do we know, what should we do? pages 485–520, 1985. (Quoted on page 40.)
- [67] William H Levison. A simulation model for driver’s use of in-vehicle information systems. *Transportation research record*, (1403):7–13, 1993. (Quoted on page 41.)
- [68] S. Data, L. Pascali, and C. Santi. Handling objective evaluation using a parametric driver model for iso lane change simulation. (*Tech. Rep. No. 2002-01-1569*), *SAE International*, 2002. (Quoted on page 42.)
- [69] U Wurster and B Schick. Substantial progress of virtual driver skills in interaction with advanced control systems to meet the new challenges of vehicle dynamics simulation. *Lecture, AVEC, Loughborough, United Kingdom*, 1(0):8, 2010. (Quoted on page 43.)

- [70] T. Bellet, P. Mayenobe, J.C. Bornard, D. Gruyer, and B. Mathern. Como-sivic: a first step towards a virtual platform for human centred design of driving assistances. In *11th IFAC/IFIP/IFORS/IEA Symposium on analysis, design and evaluation*, 2010. (Quoted on pages 45 and 47.)
- [71] D. Gruyer, C. Royere, N. du Lac, G. Michel, and J.M. Blosseville. Sivic and rt-maps interconnected platforms for the conception and the evaluation of driving assistance systems. In *Proceedings of the ITS World Congress, London, UK*, 2006. (Quoted on page 47.)
- [72] ACEA Workgroup CO2HDV. *Working document October 2011*. 2011. (Quoted on page 48.)
- [73] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003. (Quoted on page 61.)
- [74] Martin Gutlein, Eibe Frank, Mark Hall, and Andreas Karwath. Large-scale attribute selection using wrappers. In *Computational Intelligence and Data Mining, 2009. CIDM'09. IEEE Symposium on*, pages 332–339, 2009. (Quoted on page 61.)
- [75] Z. Xiao, E. Dellandrea, W. Dou, and L. Chen. Esfs: A new embedded feature selection method based on sfs. *LIRIS UMR 5205 CNRS/INSA de Lyon/Universite Claude Bernard Lyon 1/Universite Lumière Lyon 2/Ecole Centrale de Lyon, RR-LIRIS-2008-018*, Sep. 2008. (Quoted on page 61.)
- [76] A. Keh. European import has cars spinning heads, too. *New York Times*, November, 2010. (Quoted on page 73.)
- [77] P. Bernard. Une norme d'avance. *Truck & Business*, (232):20–21, 2012. (Quoted on page 74.)
- [78] CA: Google Inc Mountain View. Google earth (version 7.1.2.2041). <http://earth.google.com/>, 2013. (Quoted on page 90.)

Sommaire

1	Introduction	1
1.1	Contexte et motivation	1
1.1.1	Le contexte actuel d'économies de carburant	1
1.1.2	Le contexte du transport routier: un challenge complexe	2
1.1.3	Une spécificité du contexte poids-lourds : conduite rationnelle vs. éco-conduite	4
1.2	Les bénéfices de la modélisation durant les phases de développement des véhicules	5
1.2.1	Les moyens d'essais	6
1.3	Objectifs et problématiques	10
1.4	Approche et contributions	11
1.4.1	Approche	11
1.4.2	Contributions	12
1.5	Organisation du rapport de thèse	13
1.6	Liste des publications	14
2	Extraction de connaissances à partir de données de conduite, et modèles de conducteurs	19
2.1	Le processus d'exploration de données	17
2.2	Exploration de données à partir de données de conduite	19
2.2.1	Données de conduite	19
2.2.2	Définition des évènements de conduite	20
2.2.3	Définition des comportements de conduite	22
2.2.4	Attributs d'analyse de la conduite	25
2.2.5	Methodes d'apprentissage automatique et résultats	29
2.2.6	Choix des algorithmes d'apprentissage	31
2.2.7	Résumé	39
2.3	Etat de l'art des modèles de conducteurs	39
2.3.1	Historique	39
2.3.2	Comment un modèle de conducteur peut être adapté afin de modéliser différents comportements de conduite ?	41
3	Reconnaissance des évènements de conduite	51
3.1	Définition des évènements de conduite	52
3.1.1	Évènements <i>accélérateur=0</i>	52
3.1.2	Évènements <i>accélérateur>0</i>	54
3.2	Acquisition des données	56
3.2.1	Collecte de données pour la reconnaissance des évènements <i>accélérateur=0</i>	56

3.2.2	Collecte de données pour la reconnaissance des évènements <i>accélérateur</i> >0	57
3.3	Attributs décrivant les évènements de conduite	59
3.3.1	Évènements <i>accélérateur</i> =0	61
3.3.2	Évènements <i>accélérateur</i> >0	61
3.4	Reconnaissance des évènements à partir de données réelles avec les deux algorithmes d'apprentissage automatique	62
3.4.1	Classification des évènements <i>accélérateur</i> =0	62
3.4.2	Impact du véhicule et du conducteur sur la reconnaissance des évènements <i>accélérateur</i> =0	66
3.4.3	Classification des évènements <i>accélérateur</i> >0	66
3.4.4	Résumé	67
3.5	Reconnaissance des évènements à partir de données digitales	68
3.5.1	Définition des cycles virtuels	68
3.5.2	Base de données	68
3.5.3	Résultats	69
3.6	Conclusion	70
4	Corrélation entre le comportement de conduite et le RDI	73
4.1	Indicateur de conduite rationnelle (RDI)	74
4.2	Acquisition des données	75
4.2.1	Véhicule	75
4.2.2	Conducteurs	75
4.2.3	Route	78
4.2.4	Différences en consommation et en vitesse moyenne entre les conducteurs	78
4.3	Classification hiérarchique des conducteurs	81
4.4	Attributs de conduite rationnelle	82
4.4.1	Sur route extra-urbaine	83
4.4.2	Sur autoroute	86
4.5	Étude en simulation de l'impact des attributs de conduite à l'approche des ronds-points	89
4.5.1	Selection d'un rond-point représentatif et création d'un rond- point digital	89
4.5.2	Simulation de différents comportements de conduite en rond- point	90
4.5.3	Impact des attributs sur le RDI	92
4.6	Étude en simulation de l'impact des attributs de conduite à l'approche des péages	94
4.6.1	Simulation de différents comportements de conduite à un péage	94
4.6.2	Impact du pourcentage de lever de pied sur le RDI	95
4.7	Corrélation entre le RDI et un modèle de coût	95
4.8	Conclusion	99

5 Outil de génération de différents modèles de conducteurs	101
5.1 Structure de l'outil <i>Cycle Tuning</i>	101
5.1.1 Identification des évènements de conduite	104
5.1.2 Gestion de l'efficacité énergétique du conducteur (EEM) et identification de la dynamique du véhicule	105
5.1.3 Adapter le cycle au conducteur souhaité	105
5.1.4 Cycle d'auto-apprentissage	107
5.1.5 Systèmes de régulation au banc à rouleaux et en simulation	110
5.2 Implémentation	110
5.3 Résultats	112
5.3.1 Rond-point	112
5.3.2 Péage	112
5.3.3 Arrêt	115
5.3.4 Combinaison de plusieurs évènements	117
5.3.5 Résumé	119
6 Conclusion	121
A Les différents usages poids-lourds	125
B Liste des évènements de conduite	127
C Attributs de conduite utilisés dans l'état de l'art	129
D Documents pour la collecte de données	139
D.1 Questionnaire conducteur	139
D.2 Questionnaire accompagnateur	144
Bibliographie	147

Résumé des chapitres

Chapitre 1

Le premier chapitre présente le contexte global de ce travail. Quatre problématiques ont amené le besoin de développer des modèles de conducteurs poids-lourds plus réalistes en terme de consommation de carburant et vitesse moyenne. Tout d'abord, le coût de plus en plus important du carburant, ainsi que la tendance mondiale à la réduction de la consommation des véhicules. De plus, les conducteurs ont un impact majeur sur la consommation de carburant, encore plus important que l'impact d'un changement de véhicule. D'un point de vue constructeur, estimer dans les phases amont d'un projet les performances d'un véhicule est nécessaire. Sur les moyens que sont la simulation et le banc à rouleaux, le but est d'obtenir des résultats les plus proches possibles de la réalité. Enfin, les nouveaux véhicules et les nouveaux systèmes, comme les ADAS, nécessitent des améliorations des capacités des modèles de conducteurs existants.

Dans ce chapitre, nous introduisons également le concept de conduite rationnelle. Contrairement à l'éco-conduite, la conduite rationnelle prend en compte les impératifs dus au contexte poids-lourds. En effet, les conducteurs ont des temps de conduite réglementés et des délais de livraisons à respecter. Ils ne peuvent pas toujours se permettre de rouler moins vite, et leur efficacité se mesure donc en consommation de carburant mais également en terme de vitesse moyenne. Afin de répondre à ce besoin de modèle de conducteur plus réaliste en termes de vitesse et de consommation, nous proposons le processus suivant:

- Identifier automatiquement des événements de conduite sur des données réelles mais églament sur nos données de simulation.
- Trouver un indicateur de conduite rationnelle.
- Identifier les attributs de conduite qui différencient les types de conducteurs en terme de conduite rationnelle à l'échelle des événements.
- Proposer un système qui utilise tous les résultats précédents et qui génère des styles de conduite différents sur un parcours donné.
- Tester et évaluer les résultats en simulation et au banc à rouleaux¹.

Les challenges sont multiples, passant de la diversité des événements de conduite et de conducteurs, à la qualité et la quantité des données d'entrée. Un autre challenge est l'implémentation de nos résultats. En effet, contrairement à certains modèles

¹Les essais au banc en rouleaux sont en cours, et feront l'objet d'un rapport technique interne du groupe VOLVO

de l'état de l'art, notre modèle de conducteur doit être fonctionnel, et doit en plus s'insérer dans des outils déjà existants en tenant compte de leur limites.

Chapitre 2

Le deuxième chapitre présente une vue d'ensemble de l'état de l'art concernant l'extraction de connaissances à partir de données de conduite et des modèles de conducteurs. Après une introduction du processus d'exploration de données, nous proposons une analyse des travaux de l'état de l'art sur quatre étapes déterminantes:

- La définition des événements de conduite.
- La définition des styles de conduite.
- Les attributs de conduite utilisés dans le processus de reconnaissance des événements ou des conducteurs.
- Les méthodes d'apprentissages utilisées. Nous présentons également les deux méthodes que nous avons choisies, soit les arbres de décisions (C4.5) et la régression linéaire logistique (LLR).

Au vu de l'état de l'art, nous pouvons affirmer que tant les événements que l'application finale de nos travaux sont différents de ces études. En effet, nos événements sont décrits par l'utilisation des actionneurs (freins, accélérateur, et ralentisseurs) et nous tentons de couvrir la totalité des événements de conduite. Les autres analyses se focalisent généralement sur trois événements: accélération, freinage, virage, sans détailler quelle situation crée cette accélération ou ce virage. Les événements que nous proposons sont plus détaillés, comme par exemple rond-point, démarrage, péage, maintenir la vitesse limite, limitation de vitesse etc.

Dans un second temps, nous présentons les différents types de modèles de conducteurs dans la littérature et dans les outils de simulations, et nous détaillons les méthodes utilisées pour créer des comportements de conducteurs.

Chapitre 3

Le troisième chapitre développe la classification des événements de conduite. L'objectif est de reconnaître les situations de conduite avant d'identifier les différents comportements de conducteurs quand ils font face à ces événements. Tout d'abord nous présentons les différentes collectes de données que nous avons effectuées pour construire notre base de données.

Puis nous expliquons les étapes de la reconnaissance des événements. Une fois la base de données construite, nous avons sélectionné les attributs les plus discriminants pour la reconnaissance d'événements. Il s'avère que les attributs les plus communs sont également les plus pertinents: vitesse minimum et maximum, écart de vitesse, type de route, distance, et arrêt. Nous obtenons des résultats de classification

avec C4.5 et LLR de 74% pour les événements $accélérateur = 0$ et 95% pour les événements $accélérateur > 0$. Une fois cette classification établie, nous avons également essayé d'identifier les événements sur des parcours digitalisés (i.e., des cycles). Sur une base de données contenant des cycles urbains, extra-urbains et autoroutiers, le taux de classification se situe entre 80% et 90%.

Chapitre 4

Le quatrième chapitre concerne les différences de comportements de conduite qui impliquent des écarts en consommation et en vitesse moyenne entre les conducteurs. Tout d'abord, nous introduisons un indicateur de conduite rationnelle (le RDI) qui permet de classer les chauffeurs sur un parcours donné en fonction de leur consommation et de leur vitesse moyenne. Puis nous travaillons sur deux types d'évènements : les ronds-points sur route extra-urbaine, et les péages sur autoroute.

Pour ces deux types d'évènements, nous mesurons la corrélation entre des attributs de conduite, comme la vitesse de passage en rond-point, et le RDI. Il s'avère que le pourcentage de distance de lever de pied est un attribut déterminant pour ces deux événements, ainsi que la vitesse de passage pour les ronds-points.

Une fois cette analyse effectuée, nous utilisons notre outil de simulation pour tester différents types de comportements, traduits par ces deux attributs mentionnés ci-dessus. La simulation présente l'avantage de pouvoir comparer des résultats toutes choses égales par ailleurs, à la différence de données réelles. La simulation valide l'impact positif en termes de consommation et de vitesse moyenne de ces attributs, avec des corrélation avec le RDI allant jusqu'à 0.98.

Enfin, nous proposons également une corrélation entre le RDI et un modèle de coût adapté à ce véhicule.

Chapitre 5

Le cinquième chapitre se découpe en deux parties. Premièrement, nous introduisons la structure que nous avons développée et qui adapte un cycle à différents types de conducteurs. Puis nous prouvons que les différences de comportements implémentées actuellement dans cette structure amènent déjà des résultats significatifs en termes de consommation et vitesse moyenne en simulation.

Le principe de notre structure est de reconnaître les événements de conduite sur un cycle donné, puis de traduire les différents attributs différenciant les conducteurs en termes de consignes pour la régulation. Par exemple, pour simuler un conducteur efficace, l'outil va ajouter des consignes de lever de pied en amont des ronds-points et des péages. Ce travail implique le développement de l'outil sous Matlab, de traduire les attributs en consigne, mais également des modifications dans les systèmes de régulation actuels au banc à rouleaux et en simulation.

A la fin de ce travail de thèse, les implémentations concernent les ronds-points, les péages et les arrêts. Sur un cycle d'essai de 125km, cela représente 10% des

événements qui sont adaptés au type de chauffeur. Ces modifications ont apporté des écarts de -1.5% en consommation et +3% en vitesse moyenne entre un cycle normal et ce même cycle adapté pour un conducteur efficace.

Conclusion

Dans la conclusion de ce rapport, nous rappelons les problématiques et les besoins qui ont amené le développement de la génération de différents modèles de conducteurs en termes de conduite rationnelle. Le processus global incluait la reconnaissance d'événements de conduite, puis la corrélation entre des paramètres de conduite et le RDI sur ces événements, le tout afin d'implémenter différents comportements de chauffeurs.

Nous obtenons des taux de classification des événements suffisants pour notre application. La sélection d'attributs a permis de montrer que les paramètres suivants étaient les plus discriminants pour la reconnaissance d'événements: vitesse minimum et maximum, écart de vitesse, type de route, distance, et arrêt. Une comparaison directe avec l'état de l'art n'est pas envisageable telle que, vu les écarts entre notre définition des événements, mais également entre les données et les applications finales. Cependant, nos résultats ont été obtenus avec plusieurs méthodes, incluant C4.5 et LLR, mais également k-NN qui est une méthode couramment utilisée dans la reconnaissance d'événements.

Concernant la corrélation entre les attributs et le RDI, nous nous sommes focalisés sur les ronds-points et les péages. A notre connaissance, la corrélation d'attributs calculés sur le laps de temps d'un événement de quelques secondes seulement, avec un indicateur de conduite rationnelle qui inclut la consommation mais également la vitesse, n'avait jamais été réalisée avant. Ainsi, il n'est pas aisé de comparer nos résultats, surtout car nous travaillons dans le domaine des poids-lourds. Cependant, les résultats de corrélation sont prometteurs, et la simulation a permis de prouver et de mesurer l'impact d'un paramètre à la fois, toutes choses égales par ailleurs.

Enfin, les modifications apportées actuellement pour simuler différents modèles de conducteurs apportent des résultats satisfaisants sur la consommation et la vitesse moyenne. Sur un cycle d'essai, les modifications représentent 10% des événements, et montrent des écarts de -1.5% en consommation et de +3% en vitesse moyenne entre un cycle normal et ce même cycle adapté pour un conducteur efficace.

Malgré ces résultats, nous avons également rencontré quelques limites durant ce travail. La première est le nombre d'événements implémentés. Alors que la reconnaissance des événements couvre le maximum d'événements possible, tous ces événements ne sont pas implémentés dans l'outil. Nous avons implémenté les plus prioritaires. Dans la continuité de ce travail, il faudrait désormais analyser et implémenter les différences de comportement de conduite pour les démarrages et les accélérations.

De plus, le nombre de types de conducteurs est limité à 3 pour le moment. Il

serait intéressant de pondérer les styles de conduite, comme par exemple modéliser un conducteur *efficace mais lent*, ou *efficace et rapide*. Cependant, il faut également toujours garder à l'esprit que rajouter des types de conducteurs va également augmenter significativement le temps d'essai. Et le temps prévu pour les essais n'est pas extensible indéfiniment.

Une autre perspective qui a émergé mais qui n'a pas pu être menée à bout est l'analyse des séquences d'événements. Nous avons observé que les conducteurs efficaces utilisent intelligemment l'énergie cinétique du véhicule. En conséquence, ils freinent et ré-accélèrent moins que les autres conducteurs. Cela impacte directement le nombre d'évènements, ainsi que les séquences d'évènements. Une analyse du type chaînes de Markov pourrait révéler des différences entre les conducteurs et pourrait être implémentée en modifiant le nombre et le type des événements selon le style de conducteur.

Realistic driver behavior modeling for energy efficiency estimation during heavy-trucks vehicles development

Abstract:

In the context where fuel consumption is a growing cost center, fuel consumption of a truck coupled with its average speed is one of the key vehicle characteristics that needs to be optimized and accurately estimated during the truck design process. Consequently, we aim to create different driver behavior models for testing trucks regarding fuel consumption and average speed issues, *i.e.*, rational driving.

We propose a two-step method to model more accurately driving behavior: first, the identification of driving events through driving features. Second, the quantification of three different driving behaviors on the recognized driving events. Then we implement our results in a tool that creates these different driving behaviors. The output of this tool is a cycle adapted to a driver type in terms of fuel consumption and average speed, and that can be used in simulation and on chassis-dynamometer.

The classification of driving events reaches classification rates between 74% and 91% depending on the events. We believe that they are sufficient for our application due to the raw nature of driving events and the similarities between the different classes. We also obtain promising results concerning the correlation between driving features and rational driving index. We focus especially on typical events, namely roundabout on extra-urban roads, toll on highways and stop on urban roads.

The results of the developed tool prove to be efficient since we can now simulate different driving behaviors. On our test run in simulation, adapting only 10% of the events of a cycle produces fuel savings of 1.5% and an average speed which is 3% faster for an efficient driver than a non-efficient driver. These results are promising and we need to implement other events in the future .

Keywords: driver model, fuel consumption, simulation, driving events, machine-learning, rational driving, economical driving, driving features

Modélisation de comportements de conducteurs réalistes pour l'estimation de l'efficacité énergétique durant le développement des véhicules poids-lourds

Résumé: Dans un contexte où la consommation de carburant est un poste de coût de plus en plus important, la consommation et la vitesse moyenne d'un poids-lourd est l'une des caractéristiques clés estimées durant le développement des nouveaux véhicules. Ainsi, nous désirons créer différents modèles de conducteurs en termes de consommation et de vitesse moyenne, c'est à dire en termes de conduite rationnelle.

Nous proposons une méthode en deux étapes: premièrement la reconnaissance des événements de conduite grâce à des attributs de conduite. Puis la quantification de trois types de conduite différents sur ces événements. Suite à ces deux étapes, nous pouvons implémenter nos résultats dans un outil qui crée différents modèles de conducteurs pour la simulation et le banc à rouleaux. Les écarts entre conducteurs se mesurent en terme de consommation de carburant et de vitesse moyenne.

Le taux de classification des événements s'étend de 74% à 91% selon le type d'événements. Ces résultats sont dus à la nature même des données et aux similarités entre les classes, mais nous estimons que ces taux sont suffisants pour notre application. Nous obtenons également des corrélations prometteuses entre les attributs de conduite sélectionnés et l'indicateur de conduite rationnelle. Nous avons notamment porté notre étude sur les événements classiques: les ronds-points, les péages et les arrêts.

Les résultats de l'outil que nous avons développé sont pertinents. Nous pouvons désormais simuler différents types de chauffeurs. Sur nos essais en simulation, l'adaptation de seulement 10% des événements d'un cycle découle sur un gain en consommation de 1.5% et une vitesse moyenne 3% plus élevée pour un conducteur efficace. Ces résultats sont encourageants, surtout que le travail à venir visera à augmenter la diversité des événements couverts.

Mots clés : modèle de conducteur, consommation de carburant, simulation, événements de conduite, apprentissage, conduite rationnelle, éco-conduite, paramètres de conduite
