# Trajectory generation and data fusion for control-oriented advanced driver assistance systems

Jérémie Daniel

École Doctorale Jean-Henri Lambert ED 494

# Trajectory Generation and Data Fusion for Control-oriented Advanced Driver Assistance Systems

## PHD THESIS

defended on Wednesday, December 1$^{st}$ 2010

to obtain the title of

**Doctor of Science of the Université de Haute Alsace**

**(Specialized in Control Theory and Signal Processing)**

by

Jérémie DANIEL

**Chair members**

| | | |
|---|---|---|
| *Reviewers :* | Véronique CHERFAOUI | - Université de Technologie de Compiègne |
| | Urbano NUNES | - Coimbra University (Portugal) |
| *Examinators :* | Thierry-Marie GUERRA | - Université de Valenciennes |
| | Edouard LAROCHE | - Université de Strasbourg |
| *Thesis Advisors :* | Michel BASSET | - Université de Haute Alsace |
| | Jean-Philippe LAUFFENBURGER | - Université de Haute Alsace |

**Laboratoire MIPS / Équipe MIAM — EA 2332**

# Acknowledgments

Although it is placed at the beginning of this manuscript, these acknowledgments have been, as usual, done right at the end of my PhD. This section consequently contains everything which remains after three years of PhD: all my gratitude for the different people I have met and the last part of energy I managed to spare during the redaction of this document. This section consequently sounds like the bells of freedom for me, which may bring my mental stability back (or not). You have obviously understood that a part of these acknowledgments may contain different meanings.

This PhD has been performed in the *Modélisation Intelligence Processus Systèmes* laboratory managed by Olivier Haeberle and more precisely in the *Modélisation et Identification en Automatique et Mécanique* team, under the direction of Michel Basset and Jean-Philippe Lauffenburger. My first thanks then go to these persons who allowed me to perform this PhD in good conditions. An extrapolation (obviously using Parametric Cubic Splines) leads me to the acknowledgments for the members of the MIPS, the ENSISA and the Université de Haute Alsace.

Then I would like to warmly thank Véronique Cherfaoui, Urbano Nunes, Thierry-Marie Guerra and Edouard Laroche who have kindly accepted to give some of their precious time to review this document and to integrate my PhD chair.

Particular gratitude is also going to the following MIAM Associate professors:

- Lady Floriane Collin who has always laugh to my jokes, even if they were not funny, and who helped me a lot during the writing of this PhD.
- Benjamin Mourllion the gentleman, for his help, especially in Data Fusion . Always subtle, he was never doing some vulgar and/or bad jokes ... ho wait!
- Rodolfo Orjuela the grass eater, who have demonstrated that one can survive without eating meat. Astonishing!
- Thomas Sproesser, for everything which is referred to the oral presentation of the engineering school, thus allowing me to eat free for three weeks.
- Jean-Philippe Lauffenburger (him again) for his patience, comprehension and adaptation to my complex way of explaining things. Always looking for a solution, he was one of the major actor of this PhD.

Specials thanks are given to:

- Abderazik Birouche, the shadow hard worker who told me many times "This has no sense", thus helping me to define the quality of my work.
- Joël Lambert, for his skills in robust material design which usually involved the equipment to be heavier from several dozens of kilograms.
- Jérôme Guillet, my roommate, which managed to survive to a collocation with me without any gas mask. This may be a clue to explain why he is hardly ever happy...
- Rimyalegdo Kiebre, for its business-oriented management of the cafeteria done with a number manipulation approach. Always thinking at me when requiring some service, he is the one who teach me the true English pronunciation: *"Hello" should sound like "Rrrrrello", its my Marocan English teacher who told me!*

- Dmitry Grigoriev, for having demonstrated that assuming that another pen is better working does not automatically involve this pen to really work better.
- Sacha Bernet, who showed me that performing a PhD while having three little sons, is possible.
- Julien Caroux, Guillaume Girardin, and the other PhD students, for their help and great moments we had. I particularly thing about the farting games performed with Christophe Lamy and the different test campaign done with Gaetan Pouly.

By the way I would like to thank the huge aunt Robert, wild truck driver forever, for the different remakes of Waterworld we have done during noons at the university restaurant. More seriously, she (yes yes she's a woman) helped me a lot regarding administrative tasks.

I do not have to forget the different foreign students for their work and their help in maintaining a certain level of my English. I particularly think about Chau Truong, for the rich collaboration we had and for the good time in Vietnam.

I would like to give my warm thanks to "miss NADINE" (Newly Acquired Demonstrator for Intelligent Navigation Experiments ... or something like that) for her patience and for her cooperation during the different tests campaigns. Without her, nothing would have been possible.

Very special thanks are going to all my friends, family and especially Caroline, the most Alsacian of the Chinese girl I have ever met, who have supported me for more than eight years now. Sometimes encouraging me, sometimes not, she helped me to build my complex personality which allows me to have numerous and long discussion with my different *me*. By the way, my second *me* says that I do not have to forget *Kronenbourg*, *Fischer*, and the others for their help during these three PhD years.

*To whom it may concern.*

# Table of Contents

# List of Figures

# List of Tables

# Notations

In the following tables lie the description of the notations used in this PhD regarding the considered topic and the definition of the acronyms.

| Notation | Description | Units |
|---|---|---|
| **General mathematical notations** | | |
| $\mathbb{R}$ | Real set | |
| $\mathbb{Z}$ | Integer set | |
| $\mathbb{N}$ | Natural number set | |
| $\mathcal{P}$ | A path | |
| $M^T$ | Transpose of matrix $M$ | |
| $J(.)$ | Cost criterion to be minimized | |
| $\hat{x}$ | Estimation of $x$ | |
| $\sigma_x$ | Variance of $x$ | |
| $u$ | Control signal | |
| **Notations related to Trajectory Generation** | | |
| $t$ | Spline parameter | |
| $f_i(t)$ | Polynomial of the $i^{th}$ Spline interval regarding parameter $t$ | |
| $f_{x_i}$ | Polynomial of the $i^{th}$ Spline interval for the $x$ Cartesian coordinate | |
| $a_{f_{x_i}}$ | Coefficient $a$ of the $f_{x_i}$ Spline polynomial | |
| $X$ | $X$ Cartesian coordinate | $m$ |
| $Y$ | $Y$ Cartesian coordinate | $m$ |
| $\kappa$ | Curvature | $m^{-1}$ |
| $\dot{\kappa}$ | Curvature derivative | $m^{-2}$ |
| $R$ | Curve radius | $m$ |
| $\psi$ | Vehicle heading | $rad$ |
| $W$ | Vehicle width | $m$ |
| $L$ | Road lane width | $m$ |
| $\alpha$ | Angle described by a comparison to the $x$ Cartesian coordinate axis | $rad$ |
| $\epsilon_w$ | Inaccuracies of the digital map | $m$ |
| $VAW$ | Validity Area Width | $m$ |

| Notation | Description | Units |
|---|---|---|
| **Notations related to Longitudinal Control** | | |
| $\gamma_T$ | Lateral acceleration | $m.s^{-2}$ |
| $V_{limit}$ | Limit Speed Profile for Longitudinal Control | $m.s^{-1}$ |
| $V_{ref}$ | Reference Speed for Longitudinal Control | $m.s^{-1}$ |
| $V_{current}$ | Current vehicle speed for speed profile adaptation | $m.s^{-1}$ |
| $V_{new}$ | Desired speed for speed profile adaptation | $m.s^{-1}$ |
| $X_{LAD}$ | x Cartesian coordinate of the LAD point | $m.s^{-1}$ |
| $Y_{LAD}$ | y Cartesian coordinate of the LAD point | $m.s^{-1}$ |
| $\epsilon_v$ | Difference between the reference speed and the vehicle speed | $m.s^{-1}$ |
| **Notations related to Lateral Control** | | |
| $\phi$ | Wheel yaw angle | $rad$ |
| $r$ | Wheel yaw rate | $rad.s^{-1}$ |
| $b$ | Wheel base | $m$ |
| $v$ | Vehicle Speed | $m.s^{-1}$ |
| $x_k$ | Model state $k$ | |
| $y$ | The model output | |
| $N_p$ | Prediction horizon | |
| $N_c$ | Control Horizon | |
| $\Theta_w$ | Control signal applied to the wheel | $rad$ |
| $T_e$ | Sampling time | $s$ |
| $Q$ | Precision weight | |
| $R$ | Control weight | |
| $V_x$ | Longitudinal speed | $m.s^{-1}$ |
| $V_y$ | Lateral speed | $m.s^{-1}$ |
| $l_f$ | Distance from Cog to the front axle | $m$ |
| $l_r$ | Distance from Cog to the rear axle | $m$ |
| $CS_f$ | Front cornering stiffness | $N.rad^{-1}$ |
| $CS_r$ | Rear cornering stiffness | $N.rad^{-1}$ |
| $M$ | Vehicle mass | $kg$ |
| $J_v$ | Vehicle yaw moment of inertia | $kg.m^2$ |
| $C$ | Steering motor torque | $N.m$ |
| $U$ | Steering motor voltage | $V$ |
| **Notations related to Data Fusion and Speed Limit Assistant** | | |
| $\Theta$ | Considered frame of discernment | |
| $m_{j,i}(H_i)$ | Belief mass of element $j$ over the hypothesis $H_i$ | |
| $N_{fs}$ | Number of focal speeds | |
| $C_i$ | Criterion $i$ | |
| $Crit_v$ | Criterion numerical value | |
| $C_{v_{nav}}$ | Navigation confidence variable | |
| $C_{v_{vis}}$ | Vision confidence variable | |
| ? | Undefined speed | |

| Acronym | Signification |
|---|---|
| **AAW** | Accuracy Area Width |
| **ACC** | Adaptive Cruise Control |
| **ADAS** | Advanced Driver Assistance Systems |
| **ADASRP** | Advanced Driver Assistance Systems Research Platform |
| **bba** | basic belief assignment |
| **BS** | Braking System |
| **CAN** | Controller Area Network |
| **CC** | Cruise Control |
| **CL** | CenterLane |
| **CoG** | Center of Gravity |
| **DGPS** | Differential Global Positioning System |
| **DS** | Dempster-Shafer |
| **EH** | Electronic Horizon |
| **FF** | Forgiveness Factor |
| **FSM** | Finite State Machine |
| **GDOP** | Geometric Dilution Of Precision |
| **GLONASS** | GLObal'naya NAvigatsionnaya Sputnikovaya Sistena |
| **GNSS** | Global Navigation Satellite System |
| **GPS** | Global Positioning System |
| **HDOP** | Horizontal Dilution Of Precision |
| **LAD** | Look Ahead Distance |
| **LPV** | Linear Parameter Variant |
| **LTI** | Linear Time Invariant |
| **ML** | Multi-Level |
| **MLCP** | Most Likely Candidate Probability |
| **MPC** | Model Predictive Control |
| **ODE** | Ordinary Differential Equation |
| **PCS** | Parametric Cubic Splines |
| **PCR** | Proportional Conflict Redistribution |
| **PID** | Proportional Integral Derivative |
| **QP** | Quadratic Programing |
| **ROI** | Region Of Interest |
| **RTMaps** | Real-Time Multisensor Advanced Prototyping Software |
| **RW** | Road Width |
| **SISO** | Single Input Single Output |
| **SLA** | Speed Limit Assistant |
| **SLSR** | Speed Limit Sign Recognition |
| **V2V** | Vehicle to Vehicle |
| **V2I** | Vehicle to Infrastructure |
| **WOCM** | Without Criterion Minimization |
| **WDM** | With Distance Minimization |
| **WDRM** | With Distance to Reference Minimization |
| **WEM** | With Energy Minimization |
| **WRS** | Wheeled Rolling Systems |
| **WS** | Weighted Sum |

# Publication List

## 1 International Journal

1. J. Daniel, A. Birouche, J.P. Lauffenburger and M. Basset. *Navigation-based Constrained Trajectory Generation for ADAS.* **International Journal of Vehicle Autonomous Systems (IJVAS)** (Accepted).

## 2 International Conference With Proceedings

1. J. Daniel, J.P. Lauffenburger and M. Basset. *Multi-criterion Dempster-Shafer Fusion for Speed Limit Determination.* **18th IFAC World Congress**, Milano, Italy, August 28 - September 2, 2011 (submitted, under review).
2. J. Daniel, A. Birouche, J.P. Lauffenburger and M. Basset. *Energy Constrained Trajectory Generation for Advanced Driver Assistance Systems.* **IEEE Intelligent Vehicles Symposium (IV10)**, San Diego, CA, USA, June 21-24, 2010 (oral session).
3. J. Daniel, G. Pouly, A. Birouche, J.P. Lauffenburger and M. Basset. *Navigation-based speed profile generation for an open road speed assistant.* **12th IFAC Symposium on Control in Transportation Systems (CTS 2009)**, Redondo Beach, CA, USA, September 2-4, 2009.
4. J. Daniel, C. Truong, J.P. Lauffenburger and M. Basset. *Real-time Trajectory Generation for Advanced Driver Assistance Systems Applications.* **International Forum on Strategic Technologies (IFOST 2009)**, Ho Chi Minh, Vietnam, October 21-23, 2009.
5. A. Birouche, M. Basset, J. Daniel and J.P. Lauffenburger. *Trajectory generation combined with tracking control: a new approach.* **European Symposium of Vascular Biomaterials (ESVB 2009)**, Strasbourg, France, May 13-14, 2009.

## 3 National Conference With Proceedings

1. J. Daniel, A. Birouche, J.P. Lauffenburger and M. Basset. *Génération de trajectoires sous contraintes pour des systèmes d'aides à la conduite avancés.* **Groupe de Travail en Automatique Automobile (GTAA)**, Mulhouse, France, May 26-27, 2010.

## 4 Demonstration

1. J. Daniel, G. Pouly, A. Birouche, J.P. Lauffenburger and M. Basset. *Navigation-aided Longitudinal Control.* **Demonstration Session of the IEEE Intelligent Vehicles**

**Symposium (IV08)**, Eindhoven, Netherlands, June 4-6, 2008.

# General Introduction

The term *Mechatronics* is the synergistic combination of Mechanical engineering, Electronic engineering, Computer engineering, Control engineering, and Systems Design engineering to create, design and manufacture useful products [Dorf and Bishop, 2008]. The key elements of mechatronics are physical system modeling, sensors and actuators, signal and systems, computer and logic systems, and software and data acquisition. A mechatronic system also corresponds to a complex system on which the relations of each engineering domain are applied. This refers to the majority of the newly developed systems such as vehicles, computers, etc., which are getting more and more complex with the technological advances done upon the aforementioned engineering fields.

Mechatronic systems are usually meant to perform a set of particular tasks automatically. The control of such systems represents a great challenge. Indeed, in addition to the consideration of the system complexity, the control system has to consider elements from the mechatronic system environment to perform safely and efficiently its tasks. For instance, an aircraft autopilot has to manage the aircraft actuators regarding the information obtained simultaneously from proprioceptive sensors (i.e. sensors related to the aircraft status such as speed, acceleration, etc.) and from exteroceptive sensors (i.e. sensors related to the environment status such as wind speed, temperature, etc.). This reveals the existence of relationships which link the mechatronic system, its environment and the related controller.

At the beginning, due to technology limitations, mechatronic system controls were only performed by several and independent controllers specialized on one specific task. The global control of such systems were then processed by juxtaposing several local systems in parallel. During the last decades, the different fields of mechatronic systems have been widely studied. The development of new technologies, especially in the computer science field, allowed to radically increase the computation power. This allows real-time devices to perform more tasks simultaneously, i.e. allows to develop controllers which consider more and more aspects of the mechatronic system and its environment. These aspects can be classified into two major types:

- Constraints related to the mechatronic system or to its environment which may correspond to pre-defined rules or to limitations of the mechatronic system. These elements have to be considered by the controller to avoid situations in which the system may be in danger. For instance, a car Longitudinal Controller has to consider the vehicle deceleration capabilities to perform a safe and comfortable braking phase. This generally involves complex constrained *Controllers* which follow simple *References*.

- Information coming from the mechatronic system and its environment. Indeed, to perform complex tasks, the controller will have to define accurately the current configuration of the mechatronic system and the composition of its environment. This corresponds to a great amount of information which may exceed the information management capabilities of the controller.

The present PhD, performed in the *MIPS*[1]*/MIAM*[2] laboratory, is directly focused on these problematics by providing a constraints and an information management approach:

- The *Constraints Management* aims at dispatching the constraints over the *Controller* **and** the *Reference*. Indeed, mechatronic *System* controls are usually considering constraints only during the *Controller* synthesis step. To help in the description of the *Constraints Management* benefits, two approaches are presented in this document: the unconstrained and the constrained *Reference* generation. The first one consists in a conventional approach in which the different constraints are considered during the *Controller* synthesis step. The constrained *Reference* generation considers some of the constraints directly in the *Reference* generation.
- The focus is then placed on *Information Combination* using Data Fusion techniques. In the present context, the purpose of *Data Fusion* is to combine information originating from the *Environment* and the *System* to obtain an information of better quality and/or higher level. It corresponds to a multi-level data fusion technique which helps to improve information quality and to detect false information. The detection of the erroneous data is essential to the improvement of the *Reference* and consequently of the actions involved by the *Controller*.

Both contributions are here applied to navigation-aided Advanced Driver Assistance Systems ($ADAS$). More precisely, the *Constraint Management* benefits are described through control-oriented $ADAS$ applications. An unconstrained trajectory generation, based on Parametric Cubic Splines ($PCS$), is firstly considered for Longitudinal Control. As no constraints are considered during the $PCS$ generation, they are integrated in the Longitudinal Controller synthesis. Contrary to this, the constrained trajectories, also based on $PCS$, directly integrate the constraints in their generation, thus allow either to reduce the number of constraints that should be considered during the Lateral Controller synthesis or to define additional conditions increasing the robustness of the controller. On the other hand, the *Information Combination* is here applied over a Speed Limit Assistant ($SLA$). The latter, gives the best speed limit to the *Driver* w.r.t. the road context. This is obtained by combining information coming from multiple sensors through a multi-level fusion approach based on the Dempster-Shafer Theory. The benefits of the proposed approaches are described through their comparison to a conventional $SLA$.

In Chapter.1 the PhD context is presented. This refers to the study of the relationships linking the *System*, the *Environment* and the *Controller*. From this study, the importance to consider the *Reference* will be shown. Indeed, this *Reference* represents the desired configuration of the *System*. Its generation has consequently to be done wisely. Nevertheless, this fourth component will directly lead to the necessity of considering a constraints management and information combination approach, which represents the major contributions of this PhD. These necessities are highlighted by considering the *Vehicle/Environment/Driver* triplet. Indeed, the study of the relationships linking the three elements of this framework will respectively show the interest of unconstrained/constrained trajectory generation and a multi-level data fusion based on the Dempster-Shafer Theory.

Chapter.2 describes the Navigation-aided $ADAS$ context in which the contributions will be developed. Several studies have shown alarming figures about road injuries and fatalities, thus

[1] Modélisation, Intelligence, Processus, Systèmes.
[2] Modélisation et Identification en Automatique et Mécanique.

revealing the necessity to help the *Driver* in his driving task. This Chapter is consequently dedicated to the presentation of these systems, their composition and their limitations. The strategy adopted to overcome these limitations are finally presented: the unconstrained and constrained trajectory generation are respectively applied on a Longitudinal and Lateral Controller, while the multi-level data fusion is applied on a Speed Limit Assistant.

Chapter.3 focuses on the presentation of the unconstrained and constrained trajectory generation. If both approaches are based on the same mathematical model - Parametric Cubic Splines - their description will reveal their differences. Indeed, if the unconstrained trajectory requires *a priori* information about the location of the trajectory, the constrained trajectory defines an area in which the trajectory is allowed to lie. The consideration of trajectory generation as an optimization problem then helps in the determination of the optimal constrained trajectory while minimizing a cost criterion. The comparison of both techniques also help to see the benefits of the constraints management approach. Indeed, the unconstrained trajectory generation implies the constraints related to the *Vehicle*, the *Driver* and the *Environment* to be considered by the *Controller* - the Longitudinal Controller. In the opposite, the consideration of the constraints directly in the trajectory generation avoids their consideration in the Lateral Controller synthesis step.

Chapter.4 is dedicated to the description of the multi-level data fusion for Speed Limit Assistance. Based on the Dempster-Shafer Theory, the approach consists in the definition of two fusion levels. First a local combination of the sensor information is performed. The main advantage of this first level lies in the integration of the sensors inaccuracies in the determination of the belief level attributed to the navigation information. In other words, the proposed solution determines the reliability level of each sensor. Moreover, this approach helps in the detection of sensor errors and then in the determination of the most appropriate information. Further to this first fusion step, the second one is dedicated to the combination of the information resulting from the local sensor fusion. Contrary to conventional approaches which commonly redistribute the conflict, the present multi-sensor fusion considers the eventual conflict as an additional information. Indeed, this allows the Speed Limit Assistant to stay undecided about the final speed limit which is given to the *Driver*. Nevertheless, in those cases, the *Florea*'s conflict redistribution operator is used to give an indicative limit speed information to the *Driver*.

In order to validate the different concepts and techniques described in this PhD, simulations and real-time tests have been performed. Chapter.5 describes the test results obtained with one of the *MIPS/MIAM* test car. First, the presentation of the results obtained with the unconstrained and constrained trajectory generation will validate the constraints management approach. Indeed, the consideration of the constraints directly in the generation of the trajectories representing the *Reference*, avoids to reach the used car limitations while preserving more efficiently the *Driver* comfort. Next to this, the benefits of the multi-level data fusion are presented. As expected, the integration of the sensor inaccuracies in the determination of the sensor belief levels allows to detect more efficiently the sensor errors and to select the information which best suits the considered road context.

Finally a summary of this PhD, describing globally the benefits and the possible improvements, is proposed.

# Chapter 1

# Mechatronic Systems Control: Constraints and Information Management

**Contents**



## 1.1 Introduction

Tracking control of mechatronic systems, considering a constrained environment, is a classic problem encountered in many application fields. Numerous research studies describe a great number of approaches which depend on the context of the application (environment and system specifications), the level of control objectives (stability, performance, etc.) regarding constraints, a priori information, etc. The resulting control strategies (non linear optimal control, etc.) are generally optimized considering local criteria (limits of the system, limitation of the effector, energy, etc.) due, for example, to the complexity of the system itself (switched systems, non linear subsystems, time delays, time varying parameters, etc.). Even today, considering the increasing complexity of systems, a global approach is still utopia (optimization considering

the whole V cycle), but transverse research works give new highly promising tools for the near future. Furthermore, the reference applied to the control loop is often calculated in a separate way. Spatial and/or temporal constraints are taken into account to generate a reference path to be followed by the system. Mathematical tools, such as Spline for example, are used according to the formulation of the problem (interpolation, approximation, etc.) and the constraints to be respected.

This Chapter is dedicated to the presentation of this scientific context. This includes the description of the relationships existing between the main elements of the mechatronic system control framework: the *Controller*, the *System* and its *Environment*. Through the study of these *information* and *constraints* exchanges, the importance of a suitable *Reference* is shown. Indeed, the *Reference* constitutes the desired configuration of the *System*, so has to be defined w.r.t. the *System/Environment* capabilities and limitations. One of the main idea is here to dispatch the constraints and the information on the *Controller* **and** *Reference*, thus corresponding to an unconstrained and constrained *reference* generation. Meanwhile, the growing number of information about the the *System* and the *Environment* may exceed the *Controller* processing capabilities. To cope with this problem, an information combination approach is proposed here. In a second time, the interests of the proposed solutions are highlighted considering the *Driver/Vehicle/Environment* triplet. In this context, Data Fusion will be used for information combination while unconstrained *vs.* constrained trajectory generation will be used for constraints management. The theory basics related to these approaches, which are used in the next Chapters, are finally given.

## 1.2 "System-Environment-Controller" Framework

### 1.2.1 Description

In a general point of view, the mechatronic system control framework can be composed of three elements which evolve, in a closed loop, as presented in Fig.1.1: the *Controller*, the *System* and the *Environment* (green boxes) [Hayes et al., 2003, Birouche et al., 2009].

The central element of this scheme is the *System*. Indeed, it is the element for which the *Controller* is designed. Interactions (information and constraints feedback) exist between the *Controller* and the *System* (cf. Section.1.2.2).

Usually, mechatronic systems are represented by models, which can be of different composition and complexity (theoretical, experimental or mixed models) regarding the available knowledge about them. These models are of great help for the synthesis of the *Controller* as they provide information about the *System* to be controlled. However, models are commonly designed regarding restrictive hypotheses. They consequently imply constraints which have to be managed by the *Controller*, and which usually have an impact on the generation of the actions to be done by the *System*.

In addition, the *System* evolves in, and acts on, a specific *Environment* which may be composed of other systems to be considered. There are obviously interactions between the *System* and its *Environment*, and as the *System's* evolution depends on its interaction with the *Controller*, there exists interactions between the *Environment* and the *Controller* (information and constraints).

The *Reference* is the input of the *Controller* representing the ideal or at least the desired output of the *System* [Dorf and Bishop, 2008]. Whatever the requirements, the *Reference* involves an evolution of the *System* via the generation of specific *System* actions by the *Controller* as described by Fig.1.2. Moreover, the determination of a reliable *Reference* requires the *Sys-*

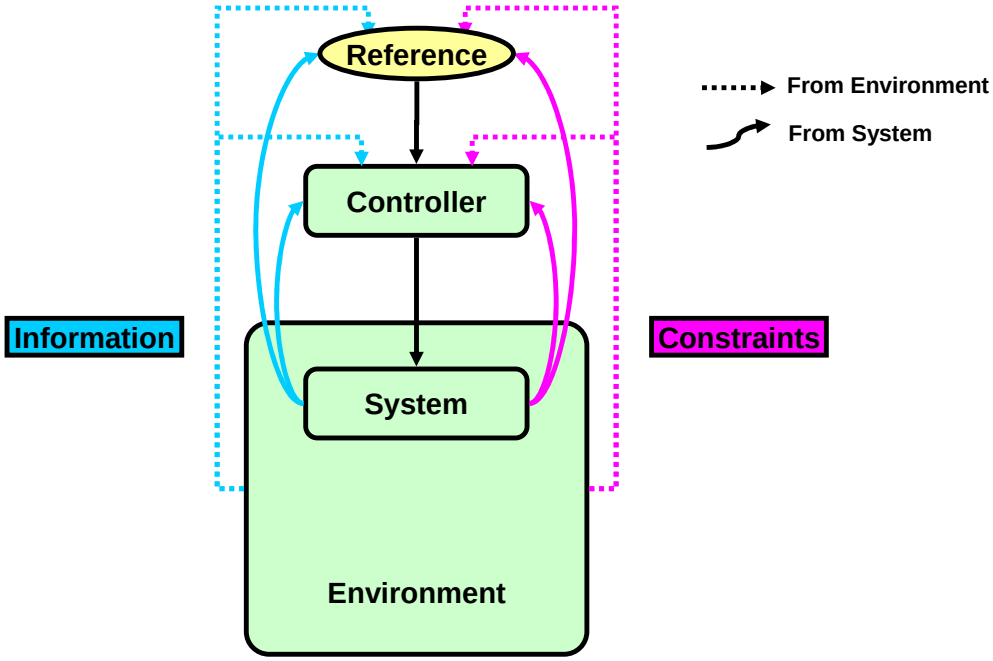



Figure 1.1: Mechatronic System Control Framework adapted from [Birouche et al., 2009]

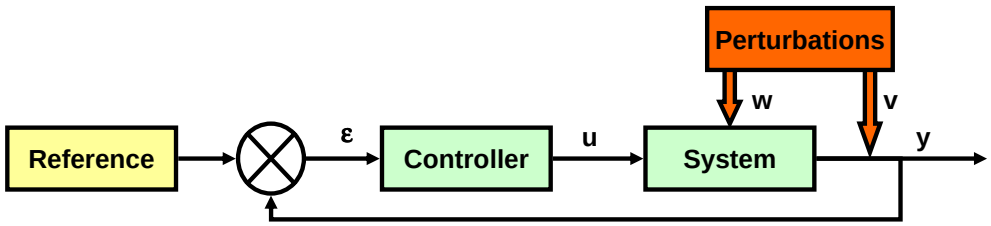



Figure 1.2: Reference as the Input of a Closed-loop System

*tem* to send back information to the *Reference*. There consequently exists *System*/*Reference* information/action relationships. In addition, the *Reference* should consider the constraints of the *System*. Then, as the *System* evolves in a specific *Environment*, the *Reference* should also consider the information and the *Environment* dependent constraints. These relationships are referred to the fact that the *Environment* feeds back information and constraints to the *Reference*.

Further to the presentation of each element of this closed loop framework, this section is dedicated to the description of these different relationships.

### 1.2.2 Notion of System

#### 1.2.2.1 Definition

A *System* is a group of interacting, interrelated, or interdependent elements forming a complex whole [Walter and Pronzato, 1997]. A *System* ($S$), can be subject to activation, deactivation, modification or reaction regarding to the signals acting on it. These signals can be classified into two types: inputs named $u$, which can be manipulated by an operator and perturbations.

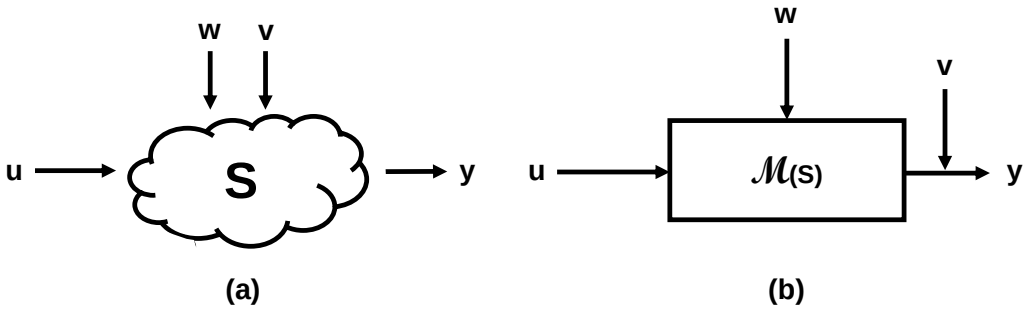



Figure 1.3: Representation of a System and a Model

Perturbations are also of two kinds: identified/measured perturbations $w$ and unknown perturbations $v$. The consequences of these inputs and perturbations are visible on the system outputs $y$. Fig.1.3.(a) describes these notions.

To be managed and controlled, a *System* is usually represented by a *Model*. Models may be designed regarding restrictive hypotheses which limit the complexity and/or the amount of parameters (in other words *System's* components) required to define the models. In addition, the unknown perturbations $v$ are usually not considered. Models are consequently defined in a certain validity frame and can be represented by Fig.1.3.(b) with $u$ the inputs, $v$ the considered perturbations and $y$ the outputs. The selection of the appropriate model is mostly determined by the available knowledge in the considered *System* [Pouly, 2009].

#### 1.2.2.2 System as a Source of Information and Constraints

The *System* represents the main source of information for the *Controller*, which may be obtained via the study of the *System's* model. Indeed, the latter allows to monitor the *System* inputs/outputs and, when they are available, its internal parameters. Having a complete view of the *System* helps in the selection of the relevant information, to be fed back to the *Controller*. This information can be of different nature (states in case of a State Space model, parameter values, etc.), depending on the requirements. For example, in the automotive domain, a car model may return information about its position, its speed, its accelerations, etc.

Almost all realistic systems (and their models) are subject to limitations which are due to their dynamics [Fraichard and Scheuer, 2004] or due to the considered models. For instance, a given *System* could be represented by several models ($LTI$, $LPV$, etc.), each of them being defined regarding their own assumptions/limitations [Biannic, 1996, Ljung, 1999]. Whatever the origin of the limitations, this restraints the set of possible configurations the *System* can take. The *System* may consequently be considered as a source of constraints which have to be considered either during the *System* modeling step or in the *Controller* synthesis step [Birouche et al., 2009].

### 1.2.3 Influence of the Environment

#### 1.2.3.1 Definition

The *Environment* corresponds to the circumstances or conditions that surround the *System* [ULT, 2010]. The *Environment* corresponds to the external elements, e.g. spatial, temporal, socio-economic, etc., which are acting on the *System*. The *System* may also have retro-actions on its *Environment*. As the *Controller* acts on the *System*, it has to consider the relationships between the pair *Environment/System*.

For example, consider a pedestrian (representing the *System* and the *Controller* simultaneously) whose purpose is to go from a point $A$ to a point $B$ using a defined in-city path. A priori, the pedestrian has only to follow the path to reach the goal. However, the pedestrian is not the lone user of the road, there may be other pedestrians or cars which can be considered as "parts" of the *Environment*. These "parts" have to be managed to avoid unwanted collisions. To reach the goal safely, the pedestrian consequently detects the elements of the *Environment* and modifies his path regarding to them. This path modification may also have a retro-action on the *Environment* by modifying the path of the other pedestrians.

#### 1.2.3.2 Environment as a Source of Additional Information and Constraints

The *Environment* is a source of information for the different elements of the mechatronic control scheme. Indeed, the example presented in the previous section explicitly shows the necessity of giving information related to the *Environment* to the pedestrian. If, in this particular case, this information helps to modify the path taken by the pedestrian, it may also be used to fulfill other purposes such as the determination of the dangerousness of the path, the type of encountered pedestrians, etc. *Environment* information also helps to define the global context of the *System* which may enhance the quality of its *Model* or its *Controller*.

The *Environment* may also restrict the evolution possibilities of the *System*. Indeed, considering the aforementioned example, the pedestrian is not allowed to walk directly on roads when sidewalks are available. This information, only available through the consideration of the *Environment*, also restricts the set of path which can be used by the pedestrians. The *Environment* then represents a source of constraints which have to be considered by the *System*, thus by the *Controller* and/or the *Reference*.

### 1.2.4 Notion of Controller

#### 1.2.4.1 Definition

A *Controller* is an interconnection of components forming a system configuration that will provide a desired response [Dorf and Bishop, 2008]. In other words, it affects the elements related to a given *System* (inputs, outputs and parameters) regarding a *Reference*. Mechatronic fields such as automotive engineering, control engineering, etc. have raised a number of optimization problems, which can be described in optimal control formulations: trajectory planning and tracking, etc. Since the years 1950-1970, the theory of optimal control has been extensively developed with powerful results, such as dynamic programming. Modern computers have made the approaches based on optimization realistic alternatives in automatic control and more precisely in tracking control.

#### 1.2.4.2 Control-based Optimization

Optimal control is one of the most important parts of control theory introduced in [Bellman, 1957] and [Pontryagine et al., 1962]. It deals with the definition of a control law which is obtained regarding an optimality criterion. This criterion commonly corresponds to a cost function which has to be minimized. For example, it may correspond to the minimization of the time required by a voltage regulator to reach a final value $V_b$ from a starting value $V_a$. This cost function usually depends on the state of the *System* and/or on control variables, and helps to define the solution using an optimization method. Optimal control presents the great advantage to allow

the integration of constraints applied to the control variables and/or to the *System* states. This globally describes the notion of *Controller*.

To illustrate this technique, let us consider a dynamic continuous-time *System* whose states are described by the solution of the following ordinary differential equation ($ODE$):

$$\frac{dx\left(t\right)}{dt} = f\left(x\left(t\right), u\left(t\right), t\right), \; x\left(0\right) = x_0 \tag{1.1}$$

Where $f$ is a $C^1$ function defined in $\mathbb{R}^{n+m+1} \rightarrow \mathbb{R}^n$, $x(t) \in \mathbb{R}^n$ the $n$-dimensional state-vector with $n$ number of states, $x_0$ the initial conditions, $u(t) \in \mathbb{R}^m$ the $m$-dimensional control inputs with $m$ number of inputs and $t \in \mathbb{R}^+$ the index time. This system can be subject to state ($x(t)$) constraints and/or control variables ($u(t)$) constraints which can be of different types such as equality, inequality, etc. Let the constraints be defined by the following expression:

$$F\left(x\left(t\right), u\left(t\right), t\right) \leq 0 \tag{1.2}$$

Finally consider a cost function $J_\infty$ defined regarding the *System* states $x(\tau)$ and the control variables $u(\tau)$ with a variable $\tau$ such as:

$$J_\infty = \int_0^\infty L\left(x\left(\tau\right), u\left(\tau\right), \tau\right) d\tau \tag{1.3}$$

with 0 the initial time, $\infty$ the final time and $\tau$ the integration variable of a function $L$.

The aim of the Optimal Control is then to find the optimal control variables $u^*(.)$ which minimizes the continuous-time cost function (1.3) while satisfying the system equation (1.1) and the constraints (1.2):

$$\begin{aligned} u^* = & \min_{u(.)} J_\infty \\ & w.r.t. \quad (1.1), (1.2) \end{aligned} \tag{1.4}$$

## 1.3 Constraints and Information Management

The relationships between the *System* and its *Environment* have already been studied in the mechatronic domain [Hayes et al., 2003, Chan and Chung, 2003, Nehmzowa and Walker, 2005]. These researches helped to enhance the global vision of *Controllers*, and now commonly consider the *System* and the *Environment* as a whole through the integration of their information and constraints. In the same time, the *Reference* generation techniques have also been widely studied resulting in the definition of new solutions which purpose is to integrate constraints of the *System* and the *Environment*. However, these works are usually focused on the integration of the information or constraints in one of these elements, the *Controller*, but hardly ever on both elements.

### 1.3.1 Constraints Management

Constraints are usually only considered during the *Controller* synthesis step, by determining the correct control signals regarding the *Reference*. This strategy generally involves complex constrained *Controllers*, which are hard to design, and simple *References*. In this PhD thesis, solutions to simplify the *Controller* synthesis step are proposed. They consist in a *Constraints*

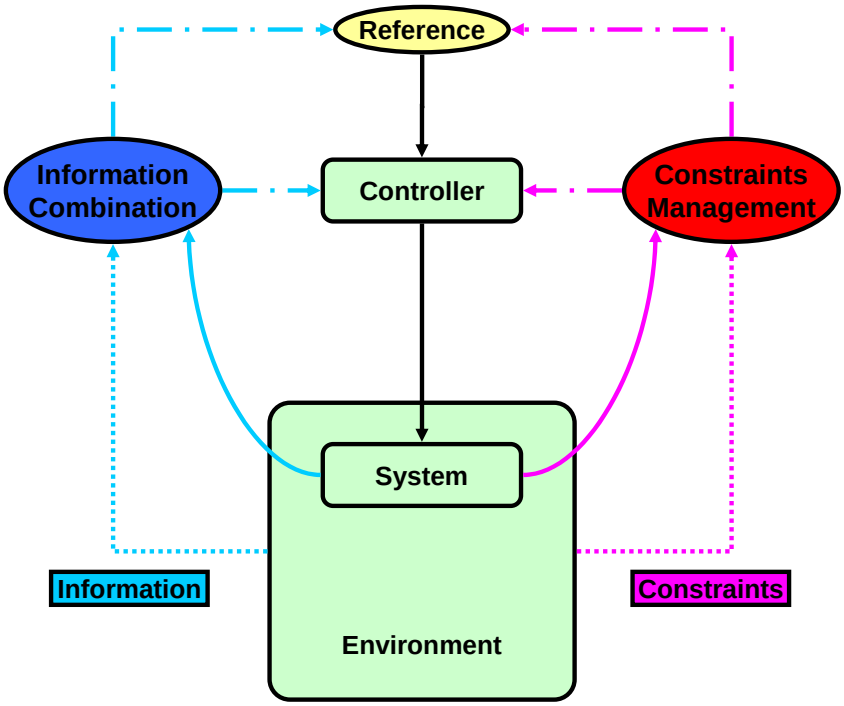



Figure 1.4: Considered General Framework

*Management* which goal is to dispatch the constraints either on the *Controller*, or on the *Reference* or even on both. To help in the description of the *Constraints Management* benefits, two approaches are presented in this document: the unconstrained *Reference* generation and the constrained *Reference* generation.

The first contribution concerns a typical approach for control-oriented applications. The spatial, temporal or physical constraints depending on the *System*, the *Environment* are mostly taken into account during the *Controller* synthesis step. This allows to define *References* with limited constraints.

The constrained *Reference* generation is based on the integration of limitations/bounds linked to the *System* and its *Environment*, at the *Reference* level. It will be shown that the selected approach, providing a control reference taking account of the identified bounds (spatial, temporal, etc.), reduces the complexity of the *Controller* synthesis step.

### 1.3.2 Information Combination

The focus is placed on *Information Combination* using Data Fusion techniques for *Information Management*. In the present context, the purpose of *Data Fusion* is to combine information originating from the *Environment* and the *System* to obtain an information of better quality and/or higher level. *Data Fusion* techniques are usually very flexible and can be applied at different levels of abstraction. For example, consider a smart home[3] equipped with sensors located in several key locations in the different rooms. A first level of data fusion also concern

[3]Smart homes refer to houses equipped with several sensors which purpose is to detect situations in which inhabitants are in danger. In critical situations, they may also launch an alarm and sent emergency signals to official corps.

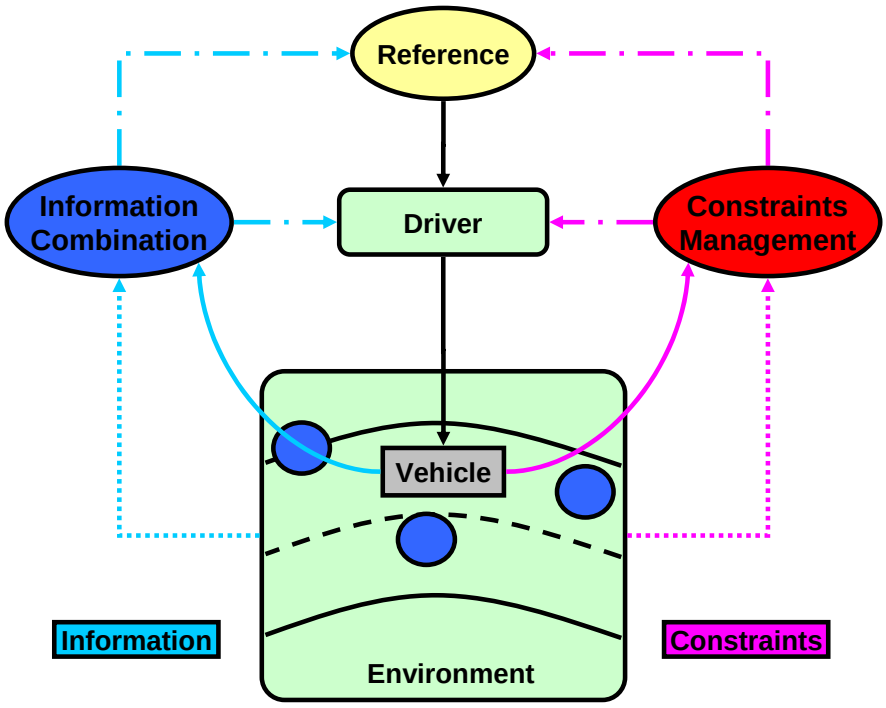



Figure 1.5: The Automotive Illustration

sensors located in the same room to determine if an inhabitant is in the room or not. Then a second level of data fusion could be the fusion of the information gathered from each room to determine the location of all the inhabitants in the house.

In the proposed work, a multi-level data fusion technique is considered. Indeed, if the example showed that multi-level fusion helps in the enhancement of the information quality, it can also be used to detect false information. The detection of the erroneous data may improve the quality of the *Reference* and consequently the actions involved by *Controller*.

## 1.4 Application to the "Driver - Vehicle - Environment" Framework

The transposition of the mechatronic control framework presented in Fig.1.4 in the automotive domain is straightforward as presented in Fig.1.5: the *System*, the *Environment* and the *Controller* are respectively replaced by the *Vehicle*, the *Enviroment* and the *Driver*. The latter is considered, at the same time, as the *Reference* generator and as the *Controller*. He is able to do this through his perception of the *Vehicle* and its *Environment*. Finally the different information/actions and constraints/actions relationships are also conserved.

This framework coincides with the previous studies of the driving task which usually consider it as a continuous evolution of the *Driver/Vehicle/Environment* triplet [Laurence, 1998, Ehmanns and Hochstadter, 2000, Lauffenburger, 2002, Glaser et al., 2002, Gruyer et al., 2005].

### 1.4.1 Framework Description

#### 1.4.1.1 The Vehicle

The *Vehicle* is the *System* which evolutes in the *Environment* and which is controlled by the *Driver*. It also represents a source of information for the *Driver*: regarding to the actions of the *Driver*, the *Vehicle* has a reaction which is sensed by the *Driver* such as visual indicators, accelerations, sounds, etc. This helps the *Driver* in his situation evaluation task, thus allowing him to adapt his actions on the *Vehicle*. However, due to its conception, the *Vehicle* is also a source of constraints. Indeed, classic vehicles are known to be non-holonomic systems: not all the solutions of the configuration space are possible and limitations in the directions of motion have to be processed [Fraichard and Scheuer, 2004]. The *Vehicle* can consequently be considered as a source of information and as a source of constraints for the *Driver*. Constraints related to the *Vehicle* will be described in Section.3.5.1 restrict the overall performance of the *Vehicle*.

#### 1.4.1.2 The Road Context

The *Road Context* is the most important component of the *Environment* in which the *Vehicle* evolutes. It can be decomposed into three elements:

- *The infrastructure geometry.* Roads are physical elements build to provide a convenient infrastructure for vehicle-based travels. Normal driving conditions imply vehicles to stay on the road, consequently *Drivers* have to take account of the geometric composition of the road: limited width, bends, etc. When stored in a database, the road geometry can be a useful source of information as the road composition may be predicted. However, the geometry is also a source of constraints as it restricts the set of possible locations for the *Vehicle*.

- *The driving rules.* If these rules may be of different natures (related to the *Vehicle*, to the road configuration, etc.), thus giving various information to the *Driver*, they all restrict the set of driving possibilities: a one-way traffic sign restricts the possible road direction to only one, etc.

- *The other components.* The last element of the *Road Context* is the presence of obstacles. Obstacles, which have to be considered as constraints to avoid accidents, can be fixed elements but can also be moving elements. This *Road Context* is consequently dynamic, contrary to the two others, and is therefore difficult to be managed. Note that other vehicles, due to the recent advances in the Vehicle-to-Vehicle communication research field, can also be considered as a source of information.

The presentation of these three elements confirms that the *Road Context* is simultaneously a source of constraints and a source of information.

#### 1.4.1.3 The Driver

The *Driver* is the *Controller* of the *Vehicle*. As each *Driver* is different, he represents an element who is very difficult to model [Lefort-Piat and Gissinger, 2002, Carsten and Cacciabue, 2007]. Nevertheless, an important element for the *Driver* is the notion of comfort[4]. Indeed, it has a

[4]By definition, it corresponds to a pleasant situation where the *Driver* does not have to do any efforts to feel good.

direct impact on the *Driver's* stress, thus on his capacity to perform the driving task correctly. In normal driving, to avoid hazardous situations, drivers are most likely to stay in a *Vehicle* configuration which is considered as safe. The measurements of the different *Vehicle* speeds and accelerations can then help to determine the *Driver's* level of comfort. That is why the comfort usually refers to the *Vehicle* accelerations, and more precisely to the lateral acceleration. A comfortable driving can consequently be characterized by a *Driver* behavior which does not involve an acceleration exceeding a predefined value [ISO, 1997, Solea and Nunes, 2006]. This last point will be considered during the constrained reference generation.

### 1.4.2 Trajectory Generation as a Reference

In mechatronic control applications and, more specifically, in the robotic field, the reference trajectory plays a key role as it gives the path which must be followed by the *System* in a safe and/or optimal way (considering obstacles to avoid and/or energy to save, for example). Multiple formulations of equations can help the engineer in the generation of a reference trajectory, depending on the application context and the objectives to reach. Up to now, most of the approaches developed have considered trajectory generation separately from tracking control design. The concept proposed here considers the generation of a reference trajectory and control design simultaneously. The idea is to directly integrate, via the reference trajectory, the geometric constraints (roadsides, continuous curvature of the trajectory, geometric characteristics of the *Vehicle*, etc.), the kinematic and the dynamic aspects of the *Vehicle*. Then, the tracking controller associated with the generated trajectory only needs to carry out the spatial and temporal controls with improved management (considering complexity, robustness and efficiency).

Considering the *Constraints Management* and the *Information Combination* approaches of this PhD, unconstrained trajectory generation implies the consideration of all the constraints at the *Controller* level, contrary to the constrained trajectory generation which simplifies the latter step by dispatching the constraints on the *Controller* **and** on the *Reference*.

#### 1.4.2.1 Trajectory Definition

The purpose of trajectory generation is to provide a set of reference points, based on a continuous mathematical model, which can be followed by a *Vehicle* satisfying several spatial/temporal constraints which can be linked to the *Vehicle*, to the *Driver* and/or to the *Environment*. Consider the example of Fig.1.6: let $q_0 = (x_0, y_0, \psi_0, \kappa_0, d\kappa_0) \in \mathbb{R}^5$ be an initial configuration defined by the *Vehicle* Center of Gravity ($CoG$)'s position ($x_0$, $y_0$), orientation ($\Phi_0$), an initial curvature ($\kappa_0$) of the path to be followed by the CoG, its respective derivative ($d\kappa_0$) and $q_n = (x_n, y_n, \psi_n, \kappa_n, d\kappa_n)$ be the desired (or final) configuration. The simplest solution to link an initial and a final configuration is the straight line. However, if this unconstrained trajectory generation solution is well suited for specific applications, it is not the case in the proposed application. Indeed, there are obstacles to be avoided (represented by large colored elements in Fig.1.6). These obstacles are constraints (herein linked to the *System's Environment*) which have to be taken into account, so revealing the necessity of *Environment* consideration.

#### 1.4.2.2 Unconstrained Trajectory Generation

Trajectory generation theory originated in the robotic domain, and more precisely on studies focused on autonomous wheeled robots. The first studies defined trajectories as a succession of straight lines and arc-circles [Dubins, 1957]. Even if this is a straightforward way to generate

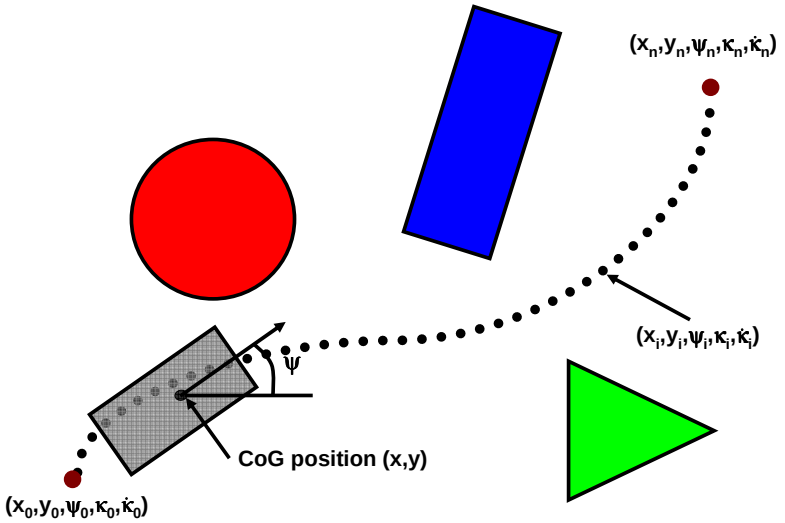



Figure 1.6: Trajectory Definition

trajectories, it has been proved that it was not suitable for wheeled robots as it does not provide curvature continuity. An improvement to this solution was found through the introduction of trajectory portions which have a polynomial curvature expressions between arc-circles and straight lines [Nagy and Kelly, 2001]. For example, in [Fraichard and Scheuer, 2004], clothoids which provide polynomial variations of curvature along their arc-length, are a good solution for junctions. However, if this method solves the problem of curvature continuity for arc-line junctions, it uses three different mathematical models whose junctions have to be correctly located and defined. More generally, trajectories which are based on polynomial curvature expressions have the advantage to directly act on the curvature representation and so, to provide at least its continuity. However, this method requires a numerical double integration of the curvature along the arc-length to get back to the trajectory coordinates. Consequently, errors are introduced into the trajectory generation process.

In the last decade, increased computer power allowed more complex mathematical models to be used. One of these models is Spline [Boor, 1978]. A Spline is a piecewise polynomial interpolation whose junctions are constrained by first and second derivatives continuities. This allows the generation of smooth curvature continuous trajectories with low degree polynomials, and low calculation time for a large number of interpolated points. Well known Spline methods are Cubic Splines and Parametric Cubic Splines [Messac and Sivanandan, 1997, Marin, 1984]. Parametrization gives more freedom to the trajectory shape as it allows the definition of multi-dimensional Splines. Another Spline, often used in the literature, is the B-Spline model [Gómez-Bravo et al., 2008]. The latter is based on the same principle than the Cubic Spline: a piecewise interpolation. The difference lies in the fact that B-Splines use Bezier curves in each interval. However, B-Splines require the definition of additional control points for all intervals. This implies an explosion of the point number which must be defined to get a trajectory.

Another mathematical model is the polar polynomial model [Nelson, 1989]. This type of polynomial is defined in a polar referential and provides curvature continuous trajectories approximating accurately circular arcs of bends. As for Cubic Splines, they use low order polynomials which are obtained according to continuity conditions on the first derivative, the slope and the curvature of the trajectory [Pinchard et al., 1996, Altafini, 1999, Lauffenburger et al., 2003]. However, the use of such polynomials requires the definition of the polar referential and trans-

Table 1.1: Unconstrained Trajectory Generation Synthesis

| *Model* | *Interests* | *Limitations* |
|---|---|---|
| Circles & straight lines | Simple mathematical expression | Not $C^2$ continuous |
| Circles & clothoids & straight lines | Simple mathematical expression, $C^2$ continuous | Definition of the junction, clothoids may introduce trajectory errors |
| Polynomial curvature curves | At least $C^2$ continuous | Trajectory computation errors (double integration) |
| Parametric Cubic Splines | Low order polynomials, $C^2$ continuous, adapted to all road context, low calculation time | Initial parameters definition |
| B-Splines | $C^2$ continuous, adapted to all road context | Definition of the control points, large number of points to be defined |
| Polar polynomials | Low order polynomials, $C^2$ continuous, good for bends | Referential definition, initial parameters pre-definition, unadapted to straight lines |

lation between polar and Cartesian coordinates. Furthermore, if polar polynomials are well suited for bends, they are not adapted to straight lines and are then often associated to quintic Cartesian polynomials [Nelson, 1989].

The trajectory generation solutions presented here are mainly based on a direct application of mathematical models (arc-circles, polynomials, etc.) which are implicitly considering a few geometrical conditions (curvature continuity, specific points location, etc.). Constraints related to the *Vehicle* and to the *Road Context* are consequently not considered. These unconstrained trajectory generation solution interests and limitations are summarized in Table.1.1. From this table, it can be seen that the unconstrained trajectory generation model which is the most interesting is the Parametric Cubic Spline ($PCS$). Indeed, contrary to the other solutions, it provides curvature continuity with low order polynomials for all road configurations (bend, straight lines, etc.) and is only limited by the selection of the initial parameters. This solution has consequently been retained in the unconstrained trajectory generation proposal.

A Cubic Spline is a piecewise polynomial interpolation, i.e. there is one polynomial for each curve linking two points (cf. Fig.1.7). However, contrary to basic interpolation methods, it avoids the use of large degree polynomials, which leads to trajectory oscillations defined as the Runge phenomenon. $PCS$ are the generalization of Cubic Spline which allows the definition of multiple dimension trajectories. This is achieved by using a parameter $t$ which divides the $PCS$ calculation into $N$ Cubic Spline calculation, $N$ being the considered dimension number. Here we consider $N = 2$ so considering two dimensional $PCS$; they are of the following form:

$$f_i(t) \begin{cases} f_{x_i}(t) = a_{f_{x_i}} t^3 + b_{f_{x_i}} t^2 + c_{f_{x_i}} t + d_{f_{x_i}} \\ f_{y_i}(t) = a_{f_{y_i}} t^3 + b_{f_{y_i}} t^2 + c_{f_{y_i}} t + d_{f_{y_i}} \end{cases} \tag{1.5}$$

with the parameter $t \in [t_i, t_{i+1}]$, for $i = 0, 2, ..., n-1$ ($n$ the number of interpolated points), $a_{f_{x_i}}$, $b_{f_{x_i}}$, $c_{f_{x_i}}$ and $d_{f_{x_i}}$ the coefficients of the $x$ Cartesian coordinate Spline, and $a_{f_{y_i}}$, $b_{f_{y_i}}$, $c_{f_{y_i}}$ and

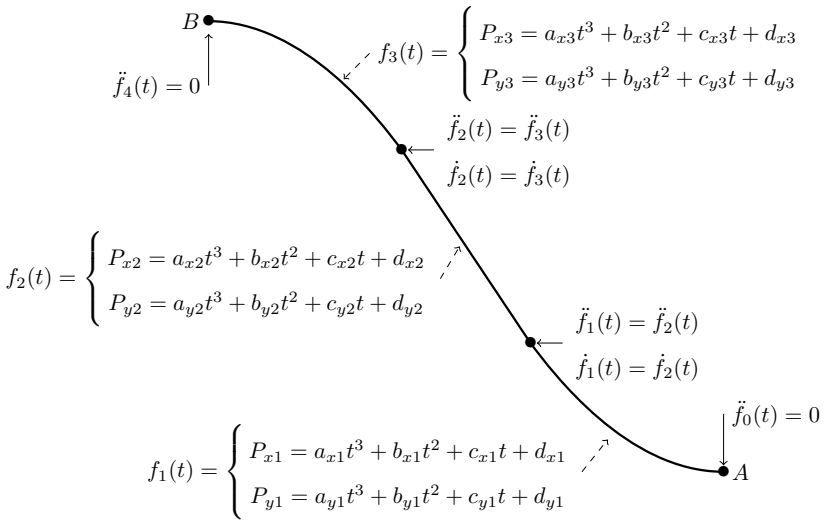



Figure 1.7: Parametric Spline Representation

$d_{f_{y_i}}$ the coefficients of the $y$ Cartesian coordinate Spline.

To ensure the smoothness of the trajectory and the continuity of the curvature, Cubic Splines are calculated under conditions: each point to be interpolated represents a position condition. Moreover, continuity conditions on the first and second derivatives of the trajectory at each point must be provided. This is obtained by solving a linear equation system which computes the second derivative values ([Boor, 1978]). However, for one set of interpolated points, an infinite number of Splines is possible. These Splines are defined depending on two major elements: boundary conditions and the expression of parameter $t$. As they can have strong effects on the Spline shape, they have to be correctly pre-defined.

- *Boundary Conditions.* In the presented solution, first and second derivative continuities were chosen in order to define a curvature continuous Spline. These continuities are provided by the Spline process for all the interpolated points except the first point $A$ and the last point $B$. As they are extreme points, they have pre-defined second derivative values. This predefinition can be performed using numerous solutions, which are known as natural, non-natural, periodic, etc. [Boor, 1978].

- *Parameter Values.* As mentioned previously, parametrization gives more freedom to the Spline. But it also adds another variable to the computation process, whose values must be chosen appropriately. The parameter definition has indeed strong effects on the trajectory shape. Most commonly used solutions to define $t$ are based on linear, chordal and centripetal expressions of the parameter [Floater, 2008].

Fig.1.8 presents the strategy adopted for the unconstrained approach build around the $PCS$ model. As constraints related to the *Vehicle* and to the *Road Context* are not considered in this approach, the inputs of the *Reference* generation is only composed of the geometric conditions defining the Spline Curve. It is very important to note that points are the only information to be considered by the $PCS$ generation. They consequently are of extreme importance as

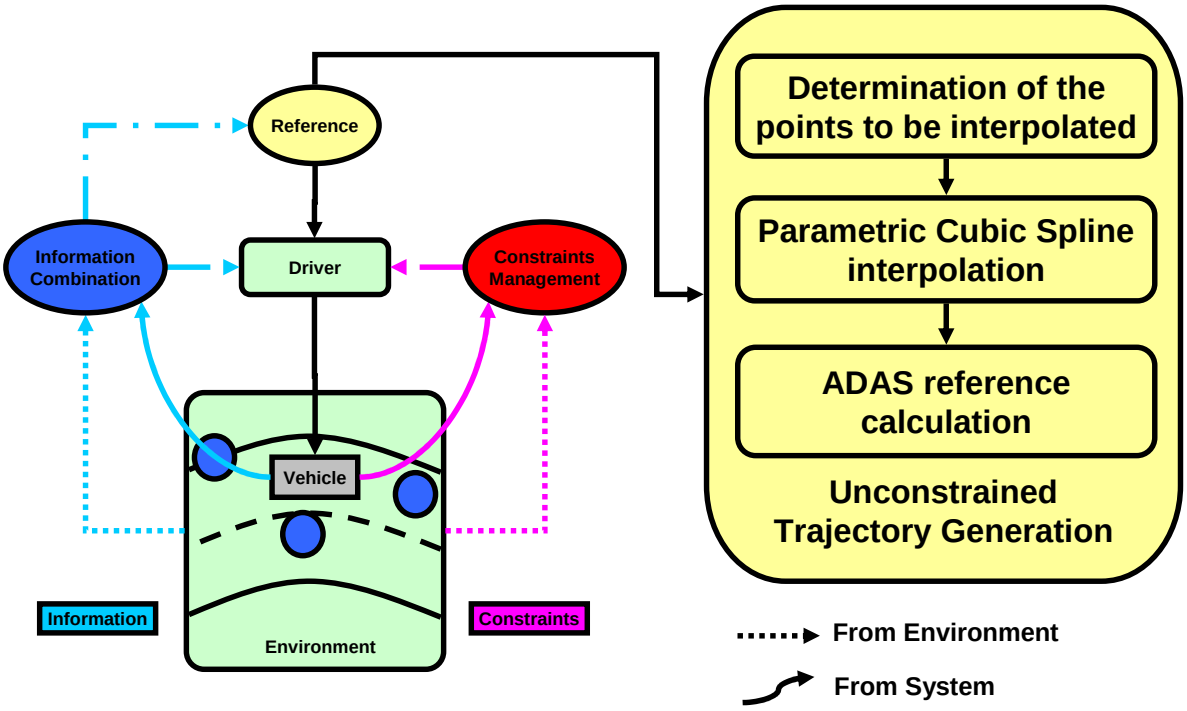



Figure 1.8: Unconstrained Trajectory Generation Strategy

they define locations which have to be included in the trajectory. The determination of these points has consequently to be done very carefully in order to avoid the generations of erroneous trajectories.

The advantages and limitations of this approach will be described through its application to the Longitudinal Control of a car-like *Vehicle* in Section.3.4.2.

#### 1.4.2.3 Constrained Trajectory Generation

For different reasons, constraints were almost always considered on the *Controller* level (during the *Controller* synthesis step), and not during the generation of the *Reference*. Usually, constrained trajectory generation are costly in terms of calculation time due to the complexity of their structure and due to the required tools (optimization techniques, precise *Vehicle* models, etc.). However, they allow to integrate additional aspects linked to the *Vehicle*, to the *Driver* and/or to the *Environment*, thus providing more realistic trajectories. Constrained trajectory generation can be classified into two categories:

- Mathematical model-based approaches: they use one of the aforementioned mathematical model. This is the case for instance in [Pinchard et al., 1996] in which kinematic and dynamic constraints linked to the *Vehicle* are integrated in a polar polynomial model. This model has been improved by [Lauffenburger et al., 2003] through the integration of constraints linked to different automotive *Driver* types (un-experimented, experimented, etc.). A similar strategy which can also be applied for trajectory generation has been presented in [Duan et al., 1999] where rational Splines are computed in a defined template (so under geometric constraints). Another approach is presented in [Egerstedt and Martin, 2001]: the Spline model is used as a basis for a trajectory generation formulated as an optimal

Table 1.2: Constrained Trajectory Generation Synthesis

| *Approach* | *Interests* | *Limitations* |
|---|---|---|
| Constrained Polar Polynomial | Low order polynomials, $C^2$ continuous, good for bends, low calculation time | Referential definition, initial parameters definition, unadapted to straight lines, considers a few constraints |
| Constrained Rational Splines | Low order polynomials, adapted to all road contexts, low calculation time | Initial parameters definition, considers a few constraints |
| Local optimized Splines | locally constrained trajectories, low order polynomials, adapted to all road contexts, average calculation time, | Initial parameters definition |
| Global optimized trajectories | Trajectories completely adapted to the considered *Vehicle* | Precise *Vehicle* model required, long calculation time, high complexity |
| Genetic Algorithms | Trajectories completely adapted to the considered *Vehicle*, | Precise *Vehicle* model required, long calculation time, high complexity |

control problem.

- The second category is not based on one of the aforementioned mathematical models. A solution could then be obtained through the use of a *Vehicle* model which "pre-tests" the trajectory as presented in [Howard and Kelly, 2007]. In this study the constrained trajectory is determined through the iteration of three steps: a trajectory is computed regarding a set of parameters which is then used by a motion predictor. The latter simulates the trajectory using a *Vehicle* model and checks if the constraints are fulfilled. If not, the parameters of the trajectory are modified and the process restarts until the requirements are met. Contrary to this solution, which corrects the trajectory after the simulation, thus generating a set of trajectories, [Bevan et al., 2010] presents a solution based on *Vehicle* model equations, which directly integrates the constraints in a multi-stage convex optimization process. This optimization algorithm minimizes a cost criterion with respect to the constraints in order to generate only one trajectory, if it exists. A similar approach is available in [Orfila, 2009]: *Vehicle* model equations are used to determine the trajectory through the use of a genetic algorithm.

The study of the interests and limitations of these two approaches, presented in Table.1.2, reveals that the second category provides trajectories which are more suited to the considered *Vehicle*. Indeed, as they are based on the prediction of the *Vehicle* configuration (using models), they define trajectories which may consider most of its aspects (steering capabilities, etc.). However, the use of *Vehicle* model is eager in calculation time which may not coincide with real-time requirements.

Contrary to this, the approach based on the integration of constraints into a mathematical model, are usually not providing trajectories which are totally suiting the considered *Vehicle*. However, they do not require large calculation time nor the consideration of the prediction step, thus may fulfill the real-time specifications. They consequently represents a good compromise

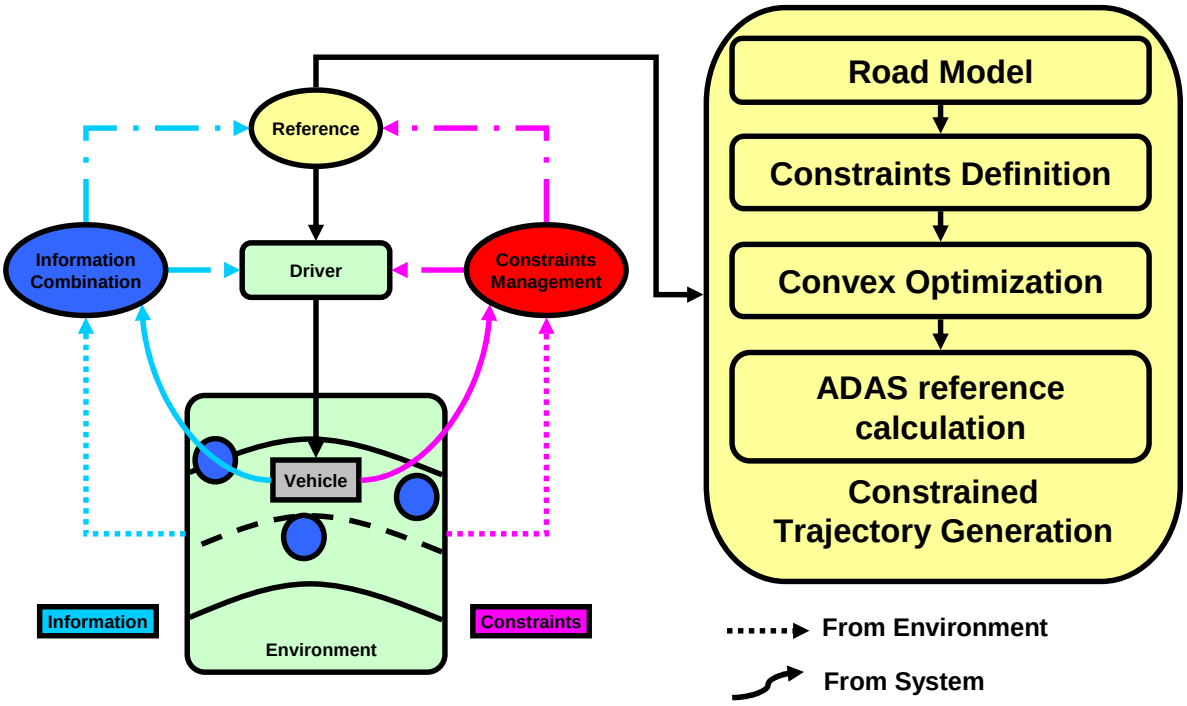



Figure 1.9: Constrained Trajectory Generation Strategy

between precision and performance, especially the Optimal Splines which seems to be well suited to constrained trajectory generation purposes. This will be confirmed further in this PhD by presenting a constrained trajectory generation approach based on *PCS*.

The adopted strategy, build around the *PCS* model, is presented in Fig.1.9. The latter, contrary to the unconstrained solution, uses information **and** constraints related to the *Vehicle* and to the *Environment*. In addition, the approach used for the trajectory generation is also completely different from the unconstrained strategy: in the unconstrained strategy the *PCS* model interpolates pre-defined points. On the opposite, the constrained trajectory generation does not consider fixed location for points, but limit boundaries. These boundaries, determined using the parameters of the road, help to integrate the problems linked to the considered application, i.e. the inaccuracies of *Digital Map Databases* (cf. Section.2.3.2) or *Vehicle* positioning errors. Finally, considering the road parameters and the constraints, an optimization algorithm is used to determine the optimal trajectory. This constrained trajectory generation is compared to the unconstrained solution and validated through its application to the Lateral control of a car-like *Vehicle* in Section.5.4.

### 1.4.3 Data Fusion for Information Enhancement

In the automotive domain, information is usually obtained from sensors which are dedicated to the measurement of a particular element. This is most of the time sufficient to perform a particular task e.g. a speed sensor is enough for Cruise Control. However, if a lone sensor could be sufficient for specific tasks, it is no more the case for more complex systems such as Advanced Driver Assistance Systems (*ADAS*). Indeed, using a lone sensor does not allow the characterization of the sensor information (inaccuracy level, realness, etc.), and the determination of the global driving situation (road context, *Vehicle* configuration) (cf. Chapter.2). This

information is required by *ADAS* which will consequently have to combine information coming from multiple sensors. This represents a challenging task as sensors may provide heterogeneous, redundant, inaccurate or erroneous information. These imperfections have to be integrated during the combination to provide coherent information. Data fusion techniques and especially the Evidence theory, are well suited for this purpose. This section is dedicated to the description of the different solution of *Data Fusion*.

#### 1.4.3.1 Data Fusion Basics

The *Data Fusion* emerged in the eighties for military purposes [Waltz and Llinas, 1990] in order to manage the great amount of information coming from battlefields. In the literature, several definitions of the term *Data Fusion* can be found. In the different close-meaning definitions, the one presented in [Wald, 1999] is more generic to describe the objectives of *Data Fusion*:

**Definition** *Data fusion is a formal framework in which are expressed means and tools for the alliance of data originating from different sources. It aims at obtaining information of greater quality, the exact definition of "greater quality" will depend upon the application.*

If there are different formalisms for *Data Fusion*, they can all be described by the scheme presented in Fig.1.10 [Martin, 2005]. Fusing data, or information[5], requires four consecutive steps: Modeling, Estimation, Combination and Decision. The steps are processed using external knowledge, e.g. given by a human expert and additional information, e.g. information which helps to characterize a sensor. The description of these steps is useful to the comprehension of the *Data Fusion* principle:

- *Modeling* consists in the selection of the formalism, in other words, the way data are represented. It represents the most important step as from the selected model depends the representation quality of the knowledge provided by experts and/or by external sources. Usually, if the *Model* is not appropriate the fusion will not give satisfying results, even with rigorous *Estimation, Combination* and *Decision* steps [Smarandache and Dezert, 2009]. The *Modeling* step could be guided by additional information relative to the considered application.
- The *Estimation* is a non-systematic step. It is used in particular cases when an intermediary step to define the numerical values attributed to data is required.
- The next step is the core of the fusion process: the *Combination*. Indeed, this step gathers the information coming from the different sources, i.e. sensors, and regroups them, so performing the fusion, using a combination operator. The latter results in an information which is of the same type than the original information, thus straightforwardly interpretable. Interesting properties for combination operators are associativity and commutativity which avoids the necessity of defining a fusion order in case of multiple-sources fusion [Shafer, 1976]. Another useful property of operators may lie in the capability to manage the conflict[6].
- Finally, the *Decision* dedicated to the selection of the most relevant solution in the solution set, is performed. The choice of the *Decision* operator is conditioned by the considered approach and by the developed *Model*.

[5] A data is considered as the physical support of the information.

[6] The conflict refers to contradictory, so to incompatible information. It usually appears when an unreliable source is used (the source gives false information), when the considered hypotheses are not exhaustive, or when sources are observing different phenomenons.

This section briefly introduces the three main *Data Fusion* techniques - the Bayesian approach, the Evidence theory and the Possibility theory - which are based on two main modeling frameworks: probabilities as the basis of the Bayesian and the Evidence theory and fuzzy sets for the Possibility theory. Other approaches such as *Voting* techniques [Lam and Suen, 1997, Kuncheva et al., 2003], the *Transferable Belief* model [Smets and Kennes, 1994], the *Dezert-Smarandache Theory* [Smarandache and Dezert, 2009], etc. are not presented here.

- The Bayesian approach, based on the Probability theory, is defined in a rigorous mathematical framework [Jaynes, 2003]. In this approach, the data and their imperfections are modeled via probability distributions ("subjective approach"), or by statistical values determined by measurements ("objective approach"). It consequently requires a good knowledge of the considered application. In addition, this approach is able to provide a good representation of the data inaccuracy[7]. Nevertheless, the Bayesian approach does not provide a good representation of data imprecision[8], and may also involve ambiguous situations in which ignorance in an event and equi-probability can be confounded. Moreover, this approach only considers singletons (the solution is an unique proposition; no unions of propositions are possible) which have consequently to be exhaustive and exclusive. Finally, the Bayesian approach is not suited for conflict management.

- The Possibility theory is mostly used when there is hardly any knowledge about the considered application. Indeed, it is based on Fuzzy Sets, introduced by [Zadeh, 1965], which can easily be used to represent the information expressed by an expert through a truth value defined in a fixed interval (usually $[0, 1]$). The Possibility theory is consequently flexible, integrates the data's inaccuracies, the sources reliability, can model the ignorance and can manage the conflict. Nevertheless, the results of the Possibility theory fusion mainly depends on the selected operator which choice has to be done wisely regarding the application.

- The Evidence theory, also called Dempster-Shafer theory [Dempster, 1967, Shafer, 1976] is based on the upper and lower probabilities. This theory is based on the modeling of data through belief levels obtained using mass functions providing a good representation of the data. This theory is usually time consuming, especially for complex problems, and the definition of the different masses requires some external and expert knowledge about the application. However, the advantage of this approach is that it can manage singletons but also unions of propositions. Data imprecision, inaccuracy and conflict management are also available with this approach as well as the modeling of the ignorance. Finally, the Evidence theory gives possibility to model, thus to integrate, the reliability of the different sources.

The three main theories advantages and limitations are summarized in Table.1.3. In this document, *Data Fusion* is applied to a Speed Limit Assistant($SLA$) which is based on the fusion of information provided by a camera and a *Digital Map Database*. To perform this data fusion, the Evidence theory has been retained. Indeed, the consideration of two sensors overcomes the computation cost limitations of the Evidence theory, which has, in addition, shown interesting results in [Bradai, 2007].

[7] The inaccuracy refers to the realness level of the data provided by the source, so to a qualitative default.

[8] The imprecision refers to the non exactitude of the data provided by the source, so to a quantitative default usually referring to a numerical error.

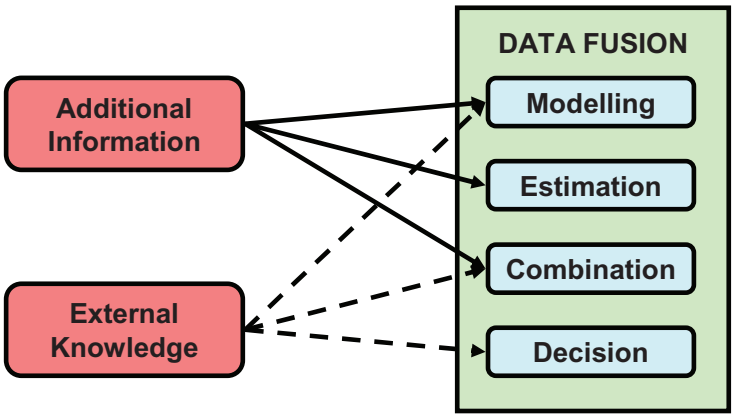



Figure 1.10: Data Fusion Scheme [Martin, 2005]

Table 1.3: Advantages and Limitations of Data Fusion Techniques

| | *Interests* | *Limitations* |
|---|---|---|
| Probability theory | Rigorous mathematical basis, inaccuracy modeling | No imprecision and ignorance modeling, only singletons, no conflict management, large knowledge about the application required |
| Evidence theory | Inaccuracy modeling, imprecision modeling, unknown modeling, source reliability modeling, conflict management, defined on subsets, good representation of data | Exponential complexity increase, knowledge about the application required |
| Possibility theory | Imprecision modeling, inaccuracy modeling, source reliability modeling, unknown modeling, conflict management, good representation of data, a few knowledge about the application required | Wise choice of the combination operator required |

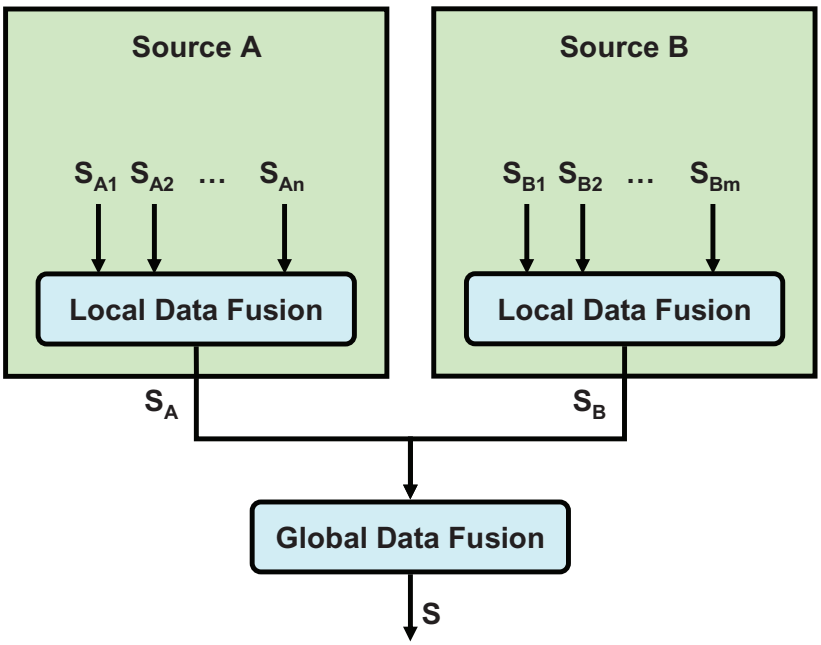



Figure 1.11: Multi-level Fusion Approach

#### 1.4.3.2 Multi-level fusion approach

A basic approach for the fusion strategy would be to directly take the information of each source and fuse them. However, sources are not exempt of defaults as they may provide imperfect information, in other words, information which can be erroneous, absent, inaccurate, imprecise, etc. The integration of the source's imperfections can be done for instance through the discounting technique. Nevertheless, this PhD is focused on a different approach: the multi-level data fusion structure.

The principle of multi-level data fusion is to divide the global fusion process into several fusion steps having different levels of abstraction as presented in Fig.1.11 [Steinberg et al., 1999, Klein, 2004]. This figure considers two levels of fusion respectively named *Local Data Fusion* and *Global Data Fusion*. The latter fuses the information obtained from source $A$ and $B$ (respectively $S_A$ and $S_B$) to determine the global information $S$. However, if the *Global Fusion Process* only sees the information coming from the sources, these information may already be the results of a preceding fusion step. This is the case in Fig.1.11, where $S_A$ and $S_B$ are determined through local fusions of sub-sources $S_{A1}$, $S_{A2}$, etc. Note that $n$ represents the total number of sub-sources for $A$ and $m$ the total number of sub-sources for $B$.

This structure has different advantages such as the discrimination of information in different levels. This is of particular help in decision taking systems such as smart homes (cf. Section.1.3.1). Indeed, in this case, several sensors describe local events ("fridge is open", "door is closed", etc.) which may be combined to determine the local situation in each room. These situations may then be combined to determine the overall house situation so that a decision can be taken. But multi-level fusion may have another advantage: it may detect source false information (cf. Section.4). This property is essential in the highly dynamic automotive systems to avoid the generation of hazardous situations.

The *Information Combination* proposed in this document is based on a two-level Evidence theory fusion with:

- The first level dedicated to the detection of the source false information. It is based on the consideration of criteria $S_{An}$ which help to characterize the sensor information quality. It will be shown in Chapter.4, that this first level of fusion helps in the detection of false information, thus enhances the quality of the information sent to the second level of fusion. In the following parts of the document, this first step is called *Multi-criterion fusion.*

- The second level dedicated to the fusion of the information coming from each source $A$ and $B$, in order to obtain the global results. In the following part of this document, this second step of fusion is called *Multi-sensor fusion.*

## 1.5 Conclusion

In this Chapter, the framework of this PhD has been presented. The latter corresponds to the *Reference/Driver/Vehicle/Environment* context which is an illustration of the global mechatronic system control framework (*Reference/Controller/System/Environment*). The description of the relationships linking these elements revealed the necessity of constraints management and information combination. Indeed, these approaches allow to dispatch the constraints and the information over the *Controller* **and** the *Reference*:

- On the one hand, this helps to simplify the *Controller* synthesis usually considering all the constraints, through the integration of some of these constraints directly in the *Reference* generation. The benefits of this approach are shown through the comparison of an unconstrained and a constrained trajectory generation approach. Both are based on $PCS$, a flexible mathematical model which is well suited for automotive applications.

- On the other hand, this allows to provide only the necessary information to the *Reference* and the *Controller*, thus avoiding these elements to process a large amount of information in addition to their original purposes. Indeed, information usually coming from sensors, may be redundant, inaccurate, erroneous, etc. so is characterized by several properties which have to be considered in the information combination. Data Fusion techniques and especially the Dempster-Shafer theory are well suited for this purpose.

The application of the presented contributions in the automotive domain is given in Chapter.2. The latter is dedicated to the presentation of the local context of this PhD: Navigation-based $ADAS$. Their composition but also their limitations as well as the strategies adopted to overcome them, are presented in this Chapter. The detailed description of the constraints management and information combination applications are finally given in the third and fourth Chapters.

# Chapter 2

# Navigation-aided Advanced Driver Assistance Systems

**Contents**



## 2.1 Introduction

In Chapter.1, it has been shown that the main contributions of this PhD correspond to unconstrained/constrained trajectory generation and to a multi-level data fusion. The present Chapter focuses on a specific field of the automotive domain: Navigation-based *ADAS* for which these contributions are dedicated.

After the presentation of the *ADAS* notion, their components, their requirements and their limitations are depicted. The strategies adopted to overcome these limitations based on the aforementioned contributions, are then provided:

- The unconstrained trajectory generation is applied on the Longitudinal Control of a car-like *Vehicle*. As no constraints are considered in the generation of the *Reference*, the strategy is here based on the consideration of constraints related to the *Vehicle*, the *Driver* and the *Environment* during the synthesis of the controller.

- The constrained trajectory generation is applied on the Lateral Control of a car-like *Vehicle*. Contrary to the unconstrained approach, here constraints are dispatched over the *Controller* and the *Reference* to reduce the controller synthesis step and to provide trajectories which are more adapted to the driving task.

- The multi-level data fusion is applied on a Speed Limit Assistant ($SLA$) which gets information from a navigation system and from a Speed Limit Sign Recognition ($SLSR$) system. As this PhD contribution refers to the improvement of an existing $SLA$, the focus of the following sections is placed on the presentation of this $SLA$ and of its limitations. Solutions, to overcome these limitations are finally introduced.

## 2.2 Navigation-aided ADAS Overview

### 2.2.1 Notion of Advanced Driver Assistance Systems

In 2004, there were **1.2 million deaths and 50 millions road injuries** [Peden et al., 2004] in the world. In addition to the human casualties, the economic cost of road accidents are estimated to be around **520 billions USD**. These figures have to be reduced! The first step for the reduction of road victims is to determine the causes of accidents. Several research studies have shown that the main source of accidents is the *Driver*: according to [Priez, 2000, Bradai, 2007] he is the source of 90% of the accidents which can occur because of a bad *Environment* perception, a false situation evaluation, an innapropriate decision, etc. Solutions to correct *Driver*'s errors are currently under intense research and one of them consists in the development of Advanced Driver Assistance Systems. Indeed, *ADAS* are designed to help the *Driver* in his driving task. To do this, *ADAS* need to gather information from the *Driver*, the *Vehicle* and the *Environment* to evaluate the current situation. When a dangerous situation is detected or predicted, *ADAS* try to help the *Driver*. This help can be of different types: it can correspond to addtional information given to the *Driver* through visual, physiological or kinestethic actuators (so corresponding to Passive safety), but can also take the control of some elements of the *Vehicle* when a critical situation occurs (so corresponding to Active safety). In conclusion, *ADAS* can be considered either as a copilot for the *Driver*, or as an entity replacing the *Driver*. This statement involves *ADAS* to be introduced in the *Driver/Vehicle/Environment* framework described in Chapter.1 such as presented in Fig.2.1. Indeed, *ADAS* may have to manage the *Vehicle* actuators, thus may act like a *Controller*.

### 2.2.2 Navigation-aided ADAS

#### 2.2.2.1 Advantages of Navigation Systems

Today, several *ADAS* are market available, e.g. *ABS*, *ESP*, parking assistance, etc. They are mainly based on proprioceptive sensors, i.e. which give information about the current *Vehicle* configuration. These sensors give local information which are usually satisfying the requirements of such *ADAS*. However, future assistance systems will require more information about the current **and** the upcoming road context [CAMP, 2004] and will have to be context aware [Feng et al., 2009]. This cannot be obtained only with these sensors. One solution can be the use of navigation systems. Indeed, they are commonly composed of a *Digital Map Database* storing a large amount of road context information. In addition, most of the navigation systems provide a process which predicts the road to be taken by the *Vehicle* [CAMP, 2004]. With the predictive data provided by navigation systems, *ADAS* may be able to reproduce the driver's

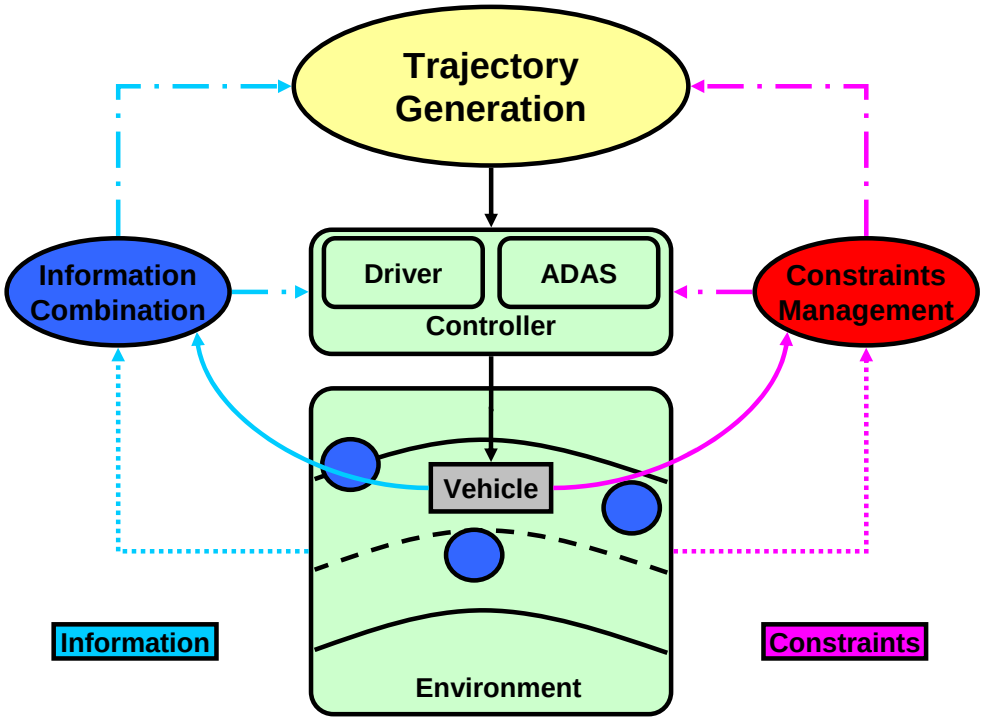



Figure 2.1: ADAS as a Component of the *Driver/Vehicle/Environment* Framework

behavior, so reproduce the different actions satisfying a safe and comfortable driving. This is of great importance for active *ADAS*.

#### 2.2.2.2 Navigation System Components

Fig.2.2 presents the common components of a navigation system: A *Receiver*, a *Digital Map Database*, a *Map-Matching Algorithm* and an *Electronic Horizon Provider*.

Navigation systems are obviously based on the use of a Global Navigation Satellite System (*GNSS*) such as *GPS*[9] or *GLONASS*[10] as they allow to estimate one's position. This position is obtained through the process of the satellite signals which is performed by the *Receiver* (in the current context, the considered *GNSS* is the *GPS*). The position is then coupled with eventual *Vehicle* information and used to determine the location of the *Vehicle* on the *Digital Map Database*. This step is done by the *Map-Matching Algorithm* which is necessary for the *Electronic Horizon Provider*. The latter uses the *Vehicle*'s location to select the relevant data in the *Digital Map Database*, so related to the set of probable roads the *Vehicle* is likely to take[11]. This relevant data, coming from the *Digital Map Database* is then accessible by *ADAS*. It is important to note that, as *Digital Map Databases* are approximate representations of the road network (cf. Section.2.3.2), they have to be used carefully. Nevertheless, the positioning step and the localization step are also of great importance (cf. Section.2.3.3).

[9] Global Positioning System.
[10] GLObal'naya NAvigatsionnaya Sputnikovaya Sistema.
[11] The depth of the Electronic Horizon is usually limited by a fixed distance value. However other limitations solution such as a fixed time window can be used.

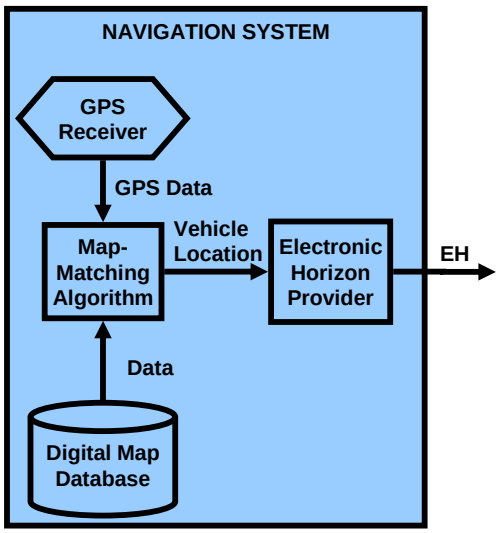



Figure 2.2: Navigation System Components

## 2.3 Information Requirements for Navigation-aided ADAS

### 2.3.1 Navigation-aided ADAS Requirements

Table.2.1 presents, in addition to current in-vehicle navigation systems, several navigation-based *ADAS* planned to be available on the market in a near, medium or long term. This table is not exhaustive but it allows having a global view of the different information needs of such kind of systems using a *Digital Map Database* as a source of information. In addition, this table clearly shows that the development of more complete *ADAS* requires an increasing number and a diversification of information.

The only market-available *ADAS* is represented by the In-vehicle Navigation. Like Near-term *ADAS* such as Intersection Warning and Curve Speed Warning [Rimenez et al., 2008], they only need digital map data with current positioning accuracy ($5m$) to attain their goal. Indeed, as they are warning oriented systems they only require little information about the current and upcoming road context. This is not the case for mid-term *ADAS* like Adaptive Front Lighting [Bradai et al., 2007, Daniel et al., 2009b] and Longitudinal Assistance [Daniel et al., 2009a] which need to be context aware (database attributes + vehicle information, etc.) to be efficient. The same point of view can be obtained through the study of long-term *ADAS* requirements. As they represent even more complex *ADAS*, they require more diversified and accurate information about the current and upcoming road lane context. In addition, they need accurate *Digital Map Database* with precise data and accurate positioning systems.

This non-exhaustive *ADAS* overview, which illustrates the requirements to be fulfilled by the next generations of navigation systems, has highlighted two major aspects:

- Future *ADAS* will require various permanent information to perform human-like assistance. This is not possible with the current reality representation of *Digital Map Databases*. Another enhancement line for navigation systems consequently lies in the determination of a solution providing permanent information for future *ADAS* such as trajectory generation or situation classification.
- Future *ADAS* will require more and more information about the road components and

Table 2.1: Example of ADAS Requirements

| *Market availability* | *ADAS* | *Data Requirements* |
|---|---|---|
| Current | In-vehicle Navigation | Database attributes + vehicle information |
| Near-term | Intersection Warning | Database attributes + vehicle information |
| | Curve Speed Warning | Database attributes, vehicle information, predictive curve speed |
| Mid-term | Adaptive Front-Lighting | Database attributes + vehicle information + permanent predictive curvatures |
| | Longitudinal Assistance | Database attributes + vehicle information + permanent predictive speeds |
| Long-term | Lane Keeping Assistance | Database attributes + vehicle information + higher localization accuracy + lane detection + permanent environment perception |
| | Autonomous driving | Database attributes + higher localization accuracy + vehicle information + permanent trajectory planning + permanent predictive curvatures + permanent predictive speeds + permanent environment perception |

Database attributes: road type, number of lanes, traffic signs, etc.
Vehicle information: speed, accelerations, yaw rate, etc.

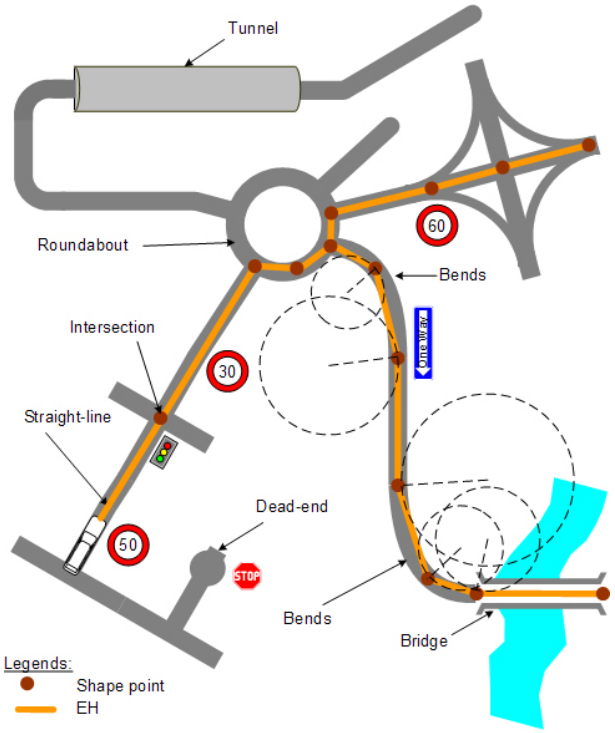



Figure 2.3: *Digital Map Database* Components Example

about the current driving lane. Consequently the accuracy and the precision of information related to the real road context have to be improved. Furthermore, the positioning and localization processes have also to be enhanced as current accuracies do not allow the determination of in-lane position and in-lane location.

### 2.3.2 Digital Map Database Limitations

Nowadays, *Digital Map Databases* contain many geographical (point coordinates, curvatures, etc.), topological (distance between points, cover time estimation between points, etc.), and informative (traffic signs, number of lanes, etc.) attributes of the real driving infrastructure. These attributes can have different levels of abstraction; for example, the number of lanes is related to a road portion while the presence of road signs is related to specific road points. Fig.2.3 presents an overview of the current database composition. The road network vectorization, based on a succession of specific points, is clearly shown. It indeed divides roads in vectors (gray lines), also called segments, linked by nodes which usually correspond to intersections. Segments can also contain shape points (brown points) giving more precise information about the current road geometry, specific attributes, etc. Finally, this figure describes the *Electronic Horizon* (*EH*) concept: it corresponds to the set of road segments the *Vehicle* is likely to take (yellow segments) in a fixed distance or time window.

If research studies on *Digital Map Databases* tend to add more and more information on them [Noyer et al., 2008], for different reasons such as navigation device memory limitations or information adding costs [CAMP, 2004], it is not possible to include all real information required for future *ADAS*. There is also a limitation in the number of attributes and points representing roads. Fig.2.4 is a snapshot of the graphical representation of a mountain bend obtained

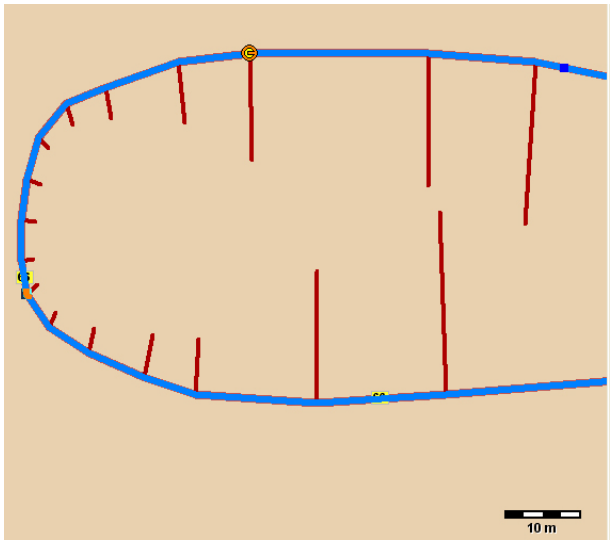



Figure 2.4: *Digital Map Database* Representation Example

from $NAVTEQ's$ navigation software: ADASRP[12] [Durekovic et al., 2007]. The current reality approximation level of the *Digital Map Databases* can be easily seen here. Indeed, it divides the curve into a succession of straight lines. These lines only correspond to the linear linking of the shape points. The orthogonal lines plotted in this figure are a graphical representation of each point's curvature. The minimal distance between two following points is around $6m$ while the mean distance is around $12m$. This discretization level can have strong effects on navigation-based *ADAS*. Autonomous driving systems which need continuous trajectories and/or continuous curvatures can not only use such digital map raw data.

In addition to the reality approximation level, *Digital Map Databases* are subject to inaccuracies. Indeed, current *Digital Map Databases* contain points having a $5m$ absolute accuracy and a $2m$ relative accuracy. As presented on the Fig.2.5 left plot, absolute errors correspond to the difference between the digitalized points and their real position on the road. In other words, a database point is located in an ellipse (gray circles) defined around the real point location (black points). The second error, the relative error, refers to the fact that database points respect the distance between each real points independently of their absolute accuracy. In the second example presented in Fig.2.5 right plot, database nodes and database points are respecting the relative accuracy even if their absolute location is totally false.

Consider the example of Fig.2.6 left plot: it shows road centerline points available in current *Digital Map Databases*. The consideration of point accuracy ellipses for this example, implies the road centerline to be located in a $5m$ width area (bounded by blue lines). The width of this "*blue area*", in which the road centerline is supposed to lie is consequently almost as large as the real road width $RW = 7m$. The latter is then used to determine the road accuracy area, i.e. the area in which the road is supposed to lie. This "*green area*", presented in Fig.2.6 right plot, possesses a width $AAW$ of $12m$, thus is almost twice the real road width. These examples clearly show the *Digital Map Database* inaccuracy level and the potential impact it may have over the *Map-Matching* step.

To improve the quality of *Digital Map Databases*, several solutions are available. The first consists obviously in the insertion of additional information into them; however, as mentioned

[12] Advanced Driver Assistance Systems Research Platform.

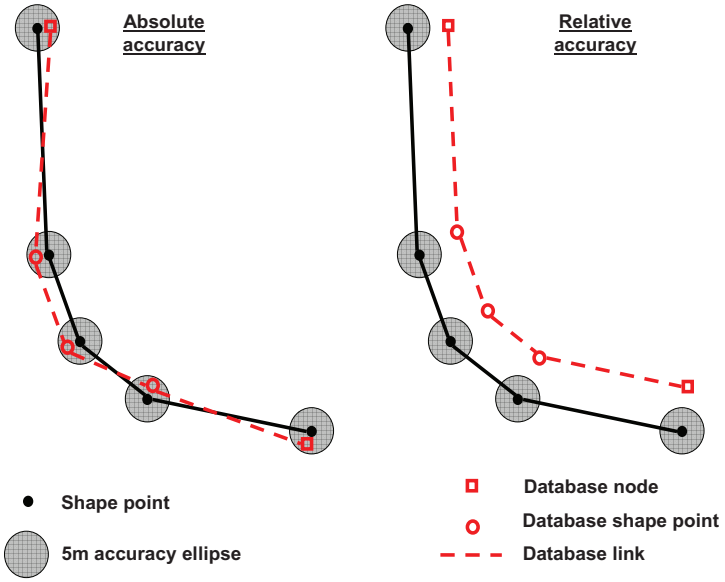



Figure 2.5: *Digital Map Database* Inaccuracy [Najjar, 2003]

previously, data additions are limited by the memory size of the navigation system devices. Another way is to use multi-level maps: the classical map which gives information of usual accuracy and a local map containing richer and more accurate information. The problem of this solution is the same as classical *Digital Map Databases*: the memory size of navigation system devices. Indeed, the accurate map data has to be stored somewhere. Research studies are currently carried out in collaboration with those dedicated to Vehicle-to-Vehicle ($V2V$) and Vehicle-to-Infrastructure ($V2I$) communication, which will allow to download, in real-time, the local and accurate data. Nevertheless, even if great advances in $V2V$ and $V2I$ communications have been done, especially through european projects such as $CVIS$[13], $COOPERS$[14] or $SAFESPOT$[15], a full operational multi-level map will not be available before several years.

### 2.3.3 Positioning and Localization Problems

It is important to note that the positioning system is also subject to errors as mentioned in [Bonnifait et al., 2008]. These errors, usually of $2m$ in the X Cartesian coordinates and $2m$ in the Y Cartesian coordinate[16] [Najjar, 2003], implies the *Vehicle* position to be included in an ellipse. Commonly, the *Vehicle* is represented by its Center of Gravity ($CoG$) which position lies in the pink ellipse in Fig.2.6 right plot. Contrary to this representation, this PhD considers the *Vehicle* to be represented by a rectangle. This involves *Vehicle* to be located in a larger area, the "*red area*" in Fig.2.6, which width may already exceeds the driving lane width $LW = 3.5m$. Consequently, it seems obvious that current navigation systems are not suited for in-lane applications, so for future $ADAS$.

As for the *Digital Map Database* and the positioning, the localization is subject to errors. Indeed, this step which consists in matching the *Vehicle* position into the *Digital Map Database*,

[13] http://www.cvisproject.org/
[14] http://www.coopers-ip.eu/
[15] http://www.safespot-eu.org/
[16] An error of $5m$ is also considered for the elevation

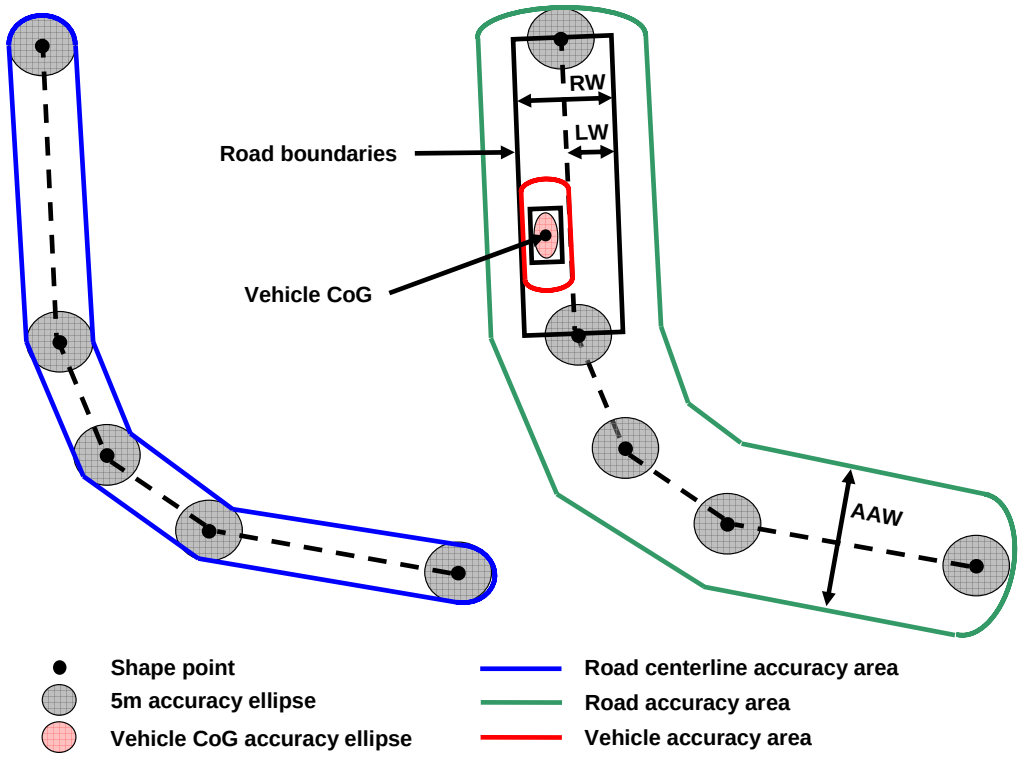



Figure 2.6: Road Inaccuracy Representation [Najjar, 2003]

may select the wrong solutions. For instance, in dense area such as cities, a *Vehicle* position may refer to several possible locations. Several solutions are available for the location selection. They are usually based on the selection of the closest database point, the closest database segment, etc. [Daniel, 2007, Najjar, 2003]. But whatever the adopted solution, the localization is still very sensitive to the quality of the positioning.

These positioning and localization problems are, as digital map data, important problems to be solved for future *ADAS*. Several research works are focused on these problems. The improvement of the positioning can be obtained through the use of a *Differential GPS* (*DGPS*). However, this solution requires a *GPS* base station which communicates with the *GPS* receiver mounted on the *Vehicle*. The communication is usually only available in a range of a few kilometers which strongly restricts the *DGPS* field of application. That is why several research studies are focusing on the enhancement of the satellite signals management or of the technique employed to combine the *GPS*, odometric and inertial information [Caron et al., 2006, Meguro et al., 2008, Kojima et al., 2008]. Even if the localization quality is dependent to the quality of the positioning and the *Digital Map Database*, improvement of the Map-Matching algorithm can be done as presented in [Jabbour et al., 2008] or [Najjar, 2003].

## 2.4 Contributions

As presented in Section.1.4, the proposed solutions are focusing on the management of the constraints and the information between the *Controller* and the *Reference*. This section is consequently dedicated to the presentation of the potential benefits of trajectory generation and data

fusion for navigation-aided *ADAS*. To demonstrate the benefits of the proposed contributions on navigation-aided *ADAS*, three of them have been retained: a Longitudinal Controller, a Lateral Controller and a Speed Limit Assistant.

### 2.4.1 Trajectory Generation for Navigation-aided ADAS

The first contribution is dedicated to the *Constraints Management* through the proposition of unconstrained and constrained trajectory generations. The aim of this section is to present the proposed solutions and their corresponding *ADAS* applications.

#### 2.4.1.1 Unconstrained Trajectory Generation for Longitudinal Control

In Section.1.4.2, it has been stated that an unconstrained *Reference* generation automatically implies the integration of all the constraints in the *Controller* synthesis step. However, unconstrained trajectory generation may already be of great help for navigation-based *ADAS* and especially for the considered *ADAS*: the Longitudinal Controller. Indeed, a study on French road accidents [Chapelon, 2008] has shown that overspeed is the second cause of accidents, so corresponds to 18.5% of the total road accidents amount. The management of the *Vehicle* speed is consequently an important element for road safety.

The most famous *Longitudinal Controller* is the *Cruise Control*: this system, when activated, maintains the speed of the *Vehicle* through its actions on the throttle angle. Another market-available *Longitudinal Controller* is the *Adaptive Cruise Control* (*ACC*) which is in pass to replace the *Cruise Control*. Its advantage is to maintain the Speed while considering the distance between the host *Vehicle* and the *Vehicle* ahead, so that, if necessary, the *Vehicle* speed is automatically decreased. However, these systems are only considering a local situation (the *Vehicle* speed and the distance with the *Vehicle* ahead), they are not using any information of the road. Several research studies have been carried out to enhance the *ACC* [Manca, 2006, Naus et al., 2008]. In [Naranjo et al., 2003], a fuzzy *Longitudinal Controller* based on information provided by a *DGPS* coupled to a reference speed database is proposed. However this reference speed database is obtained from measurements, so requires a first passage on the desired test track. Another study focuses on the determination of the best reference speed with an evolved reference speed model [Glaser et al., 2007]. The disadvantage of this system is that it requires a few parameters which are difficult to determine in real-time (road slope, road friction coefficient, etc.) or costly in terms of sensors.

In the present study, the *Longitudinal Control* approach is based on the combination of a Navigation system, a Trajectory Generation and classic *Vehicle* sensors. Indeed, one of the most interesting navigation system property is the ability to predict the road the *Vehicle* is likely to take. This can be very useful for *Longitudinal Controllers* as it allows to foresee the composition of the road. This information then helps to determine the future road context, thus allows to determine the most suited speed which has to be taken to cover the predicted road. Contrary to the aforementioned *Longitudinal Control* approaches, the proposed solution does not need a first passage on the road to be taken and does not need sensor-costly measurements.

The approach adopted here for the navigation-based *Longitudinal Control* is based on the Unconstrained trajectory generation strategy. The application of this strategy on a navigation-based *Longitudinal Controller* is then composed of (cf. Fig.2.7):

- The different elements of navigation systems (*GPS Receiver*, *Digital Map Database*, *Map-Matching Algorithm* and *Electronic Horizon Provider*), which give information about the potential road context through the *Electronic Horizon* (*EH*).

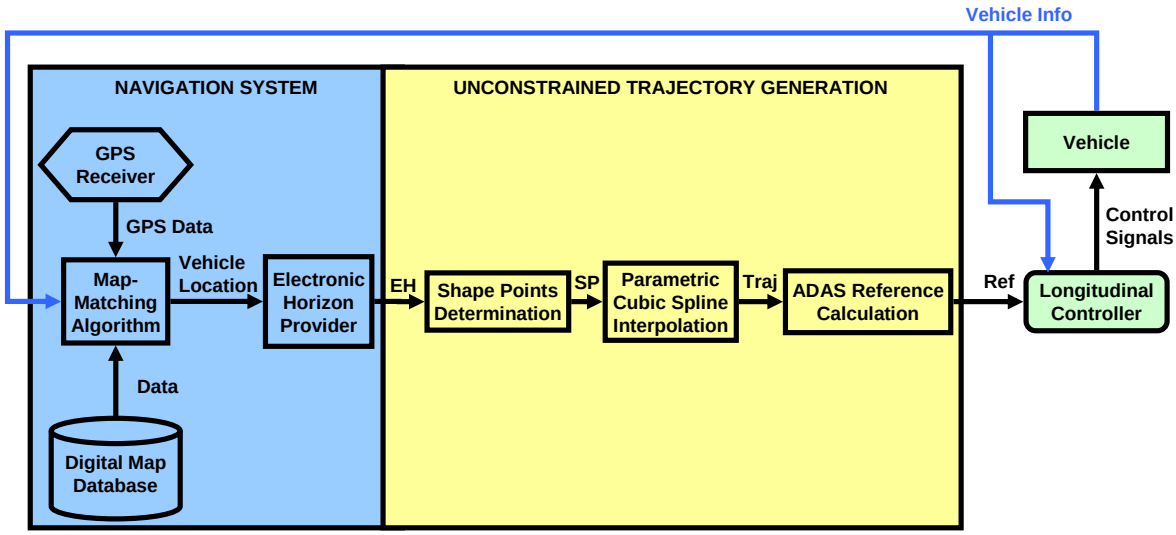



Figure 2.7: Unconstrained Trajectory Generation Approach for Longitudinal Control

- *The Shape Point Determination* which purpose is to determine the points that have to be interpolated by the Spline mathematical model. Indeed, as the *Longitudinal Controller* is here based on the unconstrained trajectory generation, the points to be interpolated have to be pre-defined. This pre-definition represents the most sensitive step of the proposed unconstrained trajectory generation. These points are defined regarding database points extracted from the *Electronic Horizon* ($EH$). To generate trajectories which are as safe as possible, middle-road points are used in combination with other digital map information to generate a set of in-lane shape points. These points ($SP$ in Fig.2.7) are then used as the basis of the *Parametric Cubic Spline Interpolation*.

- The *Parametric Cubic Spline Interpolation* purpose is to provide trajectories ($Traj$ in Fig.2.7) regarding to the set of in-lane points. These trajectories have the properties to be located in the current driving lane and to ensure the curvature continuity. Furthermore, the employed mathematical model provides a formal expression of the trajectory which helps to give the different information necessary to most of control-oriented navigation-based *ADAS*.

- The *ADAS reference calculation*. Its purpose is to use different information extracted from the trajectory to generate the information required by the Longitudinal Controller. For the considered application, this information refers to the set of predictive reference speeds ($Ref$ in Fig.2.7).

- The *Longitudinal Controller* uses the information provided by the *Vehicle* and the set of references to generate the appropriate *control signals*. The current approach is based on an unconstrained trajectory generation defined by an interpolation method. The latter is not considering any limitations which can be linked to the *Vehicle* or to the *Driver*. As these constraints are not taken into account during the generation of the *Reference*, they have to be considered during the synthesis of the *Longitudinal Controller* for a safe speed adaptation. This point is detailed in Section.3.4.2.

#### 2.4.1.2 Constrained Trajectory Generation for Lateral Control

In addition to the speed management presented in the previous section, the management of the *Vehicle* steering aspect is also of great importance. A *Lateral Controller* can be used to avoid road turnoff. Indeed, this system can be designed in such a way that it takes the *Vehicle* steering control, during eventually detected hazardous situations.

*Lateral Controllers* have been widely studied during the last decades, and several types of control are available today. Geometric solutions such as *Follow the carrot* [Hellström et al., 2006], *Pure pursuit* [Putney, 2006], *Virtual vehicle* [Egerstedt et al., 2001] or *Vector pursuit* [Wit, 2000] can be used. But there are also solutions based on *Neural Networks* [Peng et al., 2007], on *Backstepping* [Fang et al., 2005], on *Adaptive control theory* [Beji and Bestaoui, 2001], on *Fuzzy logic* [Naranjo et al., 2003, Nunes and Conde Bento, 2007], etc. Nevertheless, almost all of these *Lateral Controllers* are designed to follow a pre-defined trajectory.

In the proposed approach, the constraints are managed and dispatched on the *Controller* **but also** on the *Trajectory*. The aim of this solution is to generate trajectories integrating constraints involved by the *Road* and by the *Vehicle* which can then be used by *Lateral Controllers*. The application of this strategy on a navigation-based *Lateral Controller* is then described by Fig.2.8. The proposed contribution is composed of:

- The different elements of navigation systems (*GPS Receiver*, *Digital Map Database*, *Map-Matching Algorithm* and *Electronic Horizon Provider*), which give information about the potential road context information through the *Electronic Horizon* ($EH$).
- *The Road Model Calculation*. The *Vehicle* but also the *Environment* constraints have to be taken into account. One of the most restrictive constraint for a safe driving is that the *Vehicle* stays in its driving lane. As the considered *Lateral Controller* aims at providing a safe in-lane control, the determination of the driving lane is required. Based on the database points, the road boundaries and the driving lanes are defined.
- The *Constraints Definition*. Its purpose is to gather all the necessary information required to the formalization of the constraints related to the *Vehicle*, to the *Environment* and to the *Driver*. This formalization mainly depends on the selected optimization algorithm and on the used mathematical model.
- The *Convex Optimization*. This algorithm uses the identified constraints to generate the Spline-based constrained trajectory. In this PhD, the constrained trajectory generation is considered as an optimization problem and based on a convex optimization. Several solutions for convex optimization are available and presented in [Boyd and Vandenberghe, 2004]. Here a linear quadratic approach has been retained. The latter is based on linear constraints expressions and on a quadratic cost criterion minimization which can be straightforwardly applied to the $PCS$ model. One of this PhD contribution lies in the comparison of several cost criterion and especially to the definition of energy constrained trajectories (cf. Section.3.5.2.3). Whatever the considered cost criterion, the obtained constrained Spline-based trajectory is finally used by the *Lateral Controller* for a safe *Vehicle* steering.
- The *Lateral Controller*. With the generated trajectory, the *Lateral Controller* is able to manage the *Vehicle* actuators and consequently perform a safe steering control, through the tracking of the constrained trajectory. A Model Predictive Control ($MPC$) will be used to provide the validation results of the constrained approach.

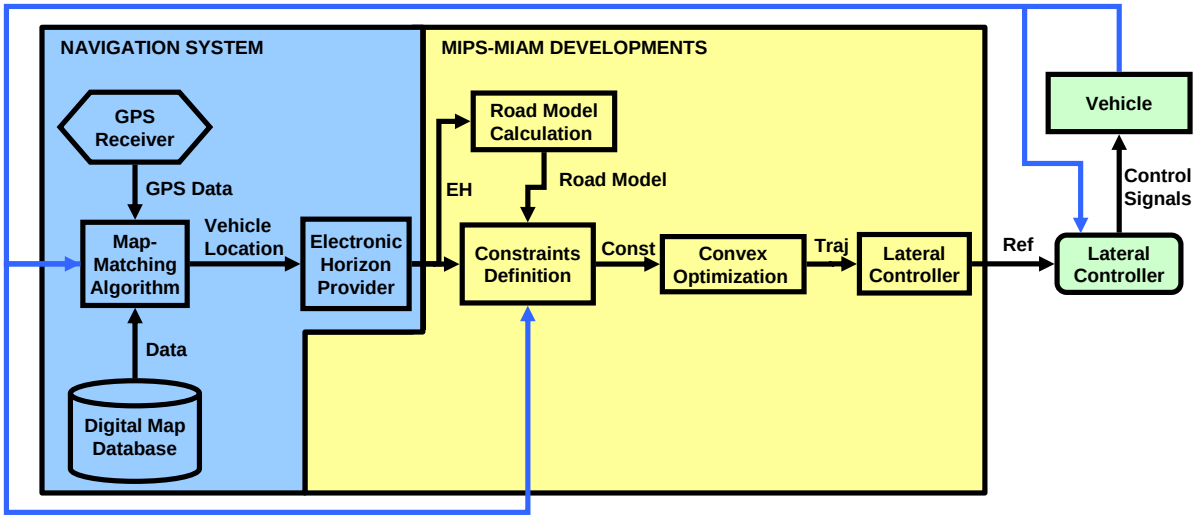



Figure 2.8: Constrained Trajectory Generation Approach for Lateral Control

### 2.4.2 Data Fusion for Speed Limit Determination

The second contribution is dedicated to the *Information Combination* through Data Fusion based on the Evidence Theory. This fusion concerns the detection of Speed limits along the road.

A study on the cause of *Driver's* errors has been done in [Chikhi, 2006] and results have shown that *Driver's* errors are mainly (50%) coming from a bad *Environment* perception. The management of the road context information is consequently a crucial element in the driving task. Indeed, the traffic is still growing [Chapelon, 2008] implying the *Driver* to treat a growing amount of information and, at the mean time, to take more, and quicker, decisions. If *ADAS* are able to manage a larger amount of information, they may still have to process heterogeneous, redundant, incacurate or erroneous information. The integration of these elements and the selection of the relevant information that have to be transmitted to the *Driver* is consequently of great importance. To show the difficulty and the complexity of information management for *ADAS*, the next sections are focused on the determination of a safe and relevant speed limit using a data fusion approach.

#### 2.4.2.1 Weighted Sum-based Speed Limit Assistant

The *SLA* proposed here is based on the fusion of the information provided by a Camera-based *SLSR* with navigation information. The proposed *SLA* is an enhancement of the system presented in [Bradai, 2007]. To highlight the limitations of the existing *SLA*, and so provide enhancement lines, the current section is dedicated to the description of the pre-existing *SLA*. The improvements of this PhD are here briefly presented and detailed in Chapter.4.

The study presented in [Bradai, 2007], proposes a data fusion approach based on the *Evidence theory*, to combine information coming from a navigation system and a camera. The goal is to improve the speed limit information quality. The principle of this fusion is presented in Fig.2.9: the *SLSR* algorithm detects in real-time the speed limit signs of the current road and evaluates a degree of confidence in the detection[17]. Simultaneously, the navigation system extracts a speed

[17]This system is dedicated to the detection of speed limit signs in Europe. These signs are as presented in Fig.2.9: circular with a red disk which contains the black numerical speed limit value.

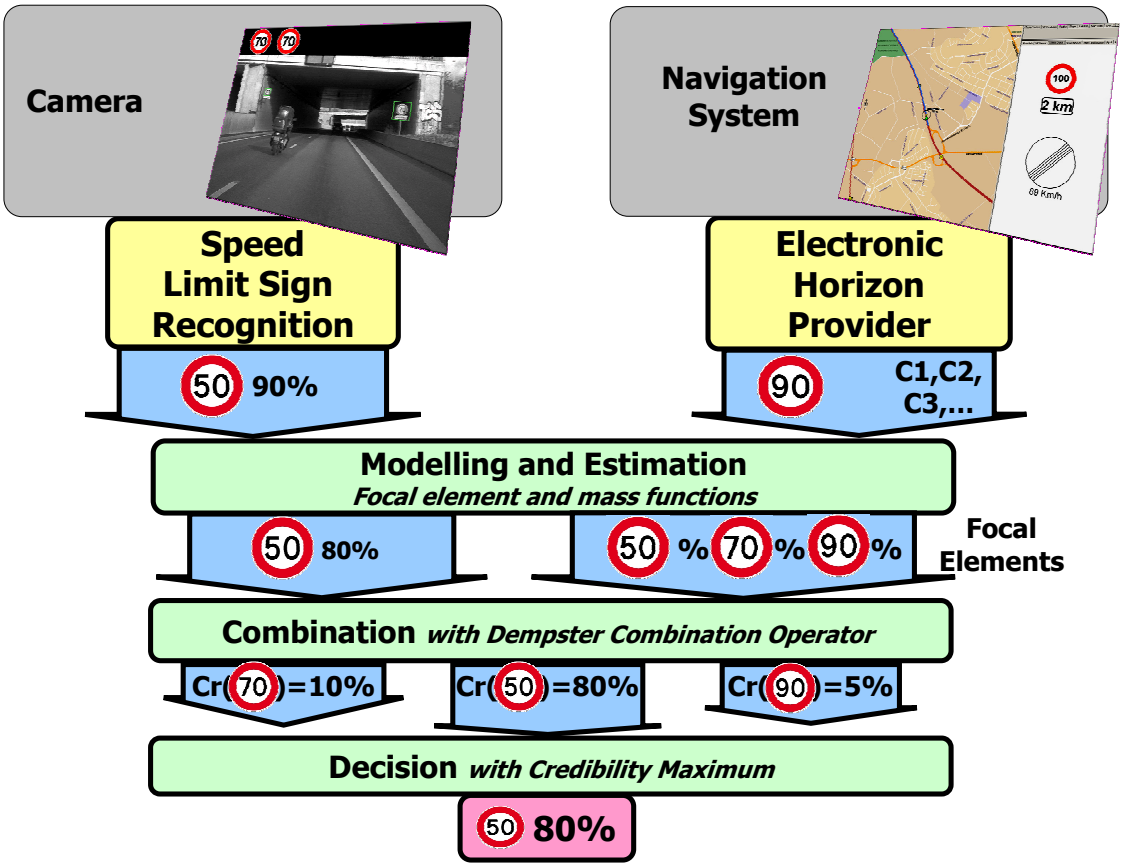



Figure 2.9: Principle of the Speed Limit Assistant [Bradai, 2007]

limit stored in the database as well as a few criteria. These criteria are used to determine the belief mass which has to be put on the navigation information and define the current driving context necessary to the fusion. Information coming from both sensors are then combined, thus resulting in the final speed limit information dedicated to the *Driver*.

**Vision Information** The information provided by the vision for the fusion is obtained through the succession of two operations: the $SLSR$ and the belief mass estimation. The $SLSR$ is performed in five steps [Bargeton, 2009]:

- The detection of circles through the Hough transform which allows to discriminate the speed limit signs,
- The transformation of these circles into squares. This helps to determine the sign Region Of Interest ($ROI$), in other words, the area which contains the numerical characters,
- A character detection is then used over the $ROI$. This process defines the location of each character in the $ROI$,
- A character recognition then aims at comparing the detected character with those stored in a database,
- If the detected character matches a database character, the $SLSR$ algorithm returns the current value with a confidence level.

The second step in the vision information determination is dedicated to the basic belief assignment (*bba*). This is done by combining the information provided by the $SLSR$, which gives the detected speed and the confidence level in this detection, with a time factor. The latter refers to the fact that the longer the sign has been detected, the smaller the confidence in this speed. Practically, this time factor is used to increase the vision uncertainty with time. Finally, the information provided by the vision is composed of a speed limit value, its belief mass which corresponds to belief in the proposition: "*It is this speed*", and a mass on the uncertainty which corresponds to belief in the proposition "*I do not know*". Note that there is no mass on the opposite element which correspond to the proposition "*It is not this speed*".

**Navigation Information** The information provided by the navigation for the fusion is obtained through the succession of two steps: the extraction of the navigation information followed by the *bba* step. The first step extracts the current speed limit value and several criteria which describes the current road context and the localization quality. The purpose of these criteria is to help in the *bba* phase, by providing a context-aware validation of the speed limit extracted from the navigation system. In addition, they should help to detect false information. Indeed, it is not rare that the navigation system returns a false speed limit; to limit the number of false information, focal elements for each speed have been defined empirically and the masses for each focal element are calculated regarding 8 criteria:

- $C_1$ refers to the localization confidence, in other words, the confidence in the selected digital road segment which has been determined by the Map-Matching algorithm further to the positioning calculated using $GPS$, odometric and inertial measurements.

- $C_2$ refers to the digitalization level of the map-matched road segment. Indeed, due to their importance (highway, country roads, etc.), roads are not digitalized with the same accuracy and with the same number of attributes. This level can be known through the checking of a specific attribute (*ADAS Attribute*) in the considered navigation system.

- $C_3$ refers to the selected road *Functional Class*. This globally refers to the road type, for example an European road has a functional class of 0 and a local road a functional class of 4.

- $C_4$ corresponds to the road nature: communal, departmental, national, highway or European.

- $C_5$, $C_6$ and $C_7$ refers to specific road situations: presence of an intersection, presence of a highway ramp or presence of a city.

- $C_8$ refers to the activation or deactivation of the navigation guidance. The term navigation guidance corresponds here to the classic in-vehicle navigation system which helps the *Driver* to go from a point $A$ to a point $B$ by determining the best path.

These criteria have been selected as, among the large amount of attributes stored in the database, their combination should allow quick and reliable determination of the current road context. In addition, they characterize the different elements of navigation systems: quality/accuracy of the digital map data ($C_2$ and $C_3$), the quality of the localization process ($C_1$), etc. These criteria are consequently of great help in the determination of the most relevant speed information.

In fact, these criteria have different values regarding to the current speed. For example in French cities, the speed is usually limited to $50km.h^{-1}$. $C_5$ which refers to the detection of in-city driving consequently give more weight in speed limits which are close or lower to $50km.h^{-1}$. Numerous empirical tests have been carried out to determine the best value for each criterion regarding to each speeds. The navigation *bba* is then performed regarding (2.1) corresponding to the mean of the coefficient value weighted sum.

$$m = \frac{\alpha_1 C_1 + \alpha_2 C_2 + \alpha_3 C_3 + \alpha_4 C_4 + \alpha_5 C_5 + \alpha_6 C_6 + \alpha_7 C_7 + \alpha_8 C_8}{\sum_{i=1}^{8} \alpha_i} \tag{2.1}$$

**Multi-Sensor Combination** In summary, at the beginning of the *Combination* step, a vision mass and a vision uncertainty mass are provided by the vision sensor, and mass for each focal element of the navigation speed are available from the navigation sensor. More details about the focal elements are available in Section.4.6.1 and in Fig.D.1. The *Combination* then consists in the computation of the information provided by both sensors. This is done through the application of the Dempster combination operator (cf. (4.10)) over the vision information and each of the navigation focal elements. The results of this combination is a belief mass attributed for each speed. Note that the Dempster combination operator is here used in a *closed world* configuration, the conflict is redistributed over the other elements via the *Dempster Normalization* technique [Shafer, 1976].

The *Decision* is performed by considering only the speed which has the maximum of belief over the different possible speeds.

#### 2.4.2.2 A multi-level fusion-based Speed Limit Assistant

The *SLA* presented in the previous section, if it gives interesting results, is not exempt of limitations on which some improvements can be brought. The main points that will be enhanced are:

- The global approach of the fusion will be modified: the multi-sensor fusion will be considered as a second fusion step of the *SLA*. The proposed approach is consequently to consider a multi-level fusion. By considering that each criteria is an independent expert/sensor, the weighted sum (2.1) can be replaced by an Evidence-theory based fusion which represents the first stage of the proposed multi-level *SLA* (cf. 4.3). This improves the accuracy and the realness of the information given to the *Driver*.

- Then the navigation *bba* will be modified. In [Bradai, 2007]'s *SLA*, the determination of the different masses is based on fixed and discrete values. Here, the *bba* will be based on a continuous representation using linear functions. These linear functions help to determine the speed mass value regarding a given confidence variable. The latter helps to characterize the confidence level which can be imputed to the navigation information (cf.4.5.4). This approach improves the flexibility of the *bba* through a dynamic criterion values determination.

  In addition the criteria number and role will be optimized. Indeed, several criteria used to describe the road context are originally aiming at characterizing the reliability of the navigation system. This is the case of $C_1$ ($MLCP$) and $C_2$ ($ADASAttribute$) which respectively characterize the quality of the localization and the quality of the digital map information. They may consequently be used in the calculation of the navigation confidence variable (cf. Section.4.5.1).

- As for the navigation information, an enhancement of the vision *bba* is proposed. To avoid aberrant fusion results, the harmonization of the *bba* step for each sensors is proposed. This implies to consider the camera detection confidence as the vision confidence variable which is used to determine its masses. The time factor, previously used to increase the uncertainty will then be used to decrease the confidence variable (cf. Section.4.5.2).

- The last aspect concerns the management of the conflict during the combination step. In the weighted sum-based *SLA*, the conflict is redistributed using the classic *Dempster Normalization* method. Here, the conflict is considered as an information in the fusion process. Indeed, its generation usually corresponds to the fact that sources are discordant, so revealing the non exhaustiveness of the discernment frame or a sensor false detection. The redistribution of the conflict may consequently be interpreted as a loss of information in the fusion process. In this document, a strategy based on the use of the conflict as a source of information is considered (cf. Section4.7)).

## 2.5 Conclusion

In this Chapter, Navigation-based *ADAS* have been presented. These systems are of great help in the improvement of driving safety as they help the *Driver* in his driving task. Nevertheless, they are still subject to limitations (sensor inaccuracies, average numerical representation of the *Digital Map Database*, etc.) which can be overcome by applying the proposed contributions. Indeed, the generation of unconstrained and constrained trajectories helps to provide relevant

data to the Lateral and Longitudinal Control of a car-like *Vehicle*, while the multi-level data fusion approach helps to provide only the relevant Speed Limit information to the *Driver*. In this Chapter, the strategies related to these works have been introduced; details about the unconstrained/constrained trajectory generation and multi-level data fusion are given respectively in Chapter.3 and Chapter.4.

# Chapter 3

# Trajectory Generation for Car-like Mechatronic Systems

**Contents**



## 3.1 Introduction

In Chapter.2, the global strategy for unconstrained and constrained trajectory generation respectively applied to the Longitudinal and Lateral Control of a car-like vehicles, have been presented. This Chapter aims at describing the different techniques adopted to perform a safe and comfortable in-lane driving:

- First, details about the unconstrained trajectory generation approach are given. This consists in the presentation of the tuning of the $PCS$ through the definition of the continuity conditions and the parameter expression. Then, as the unconstrained trajectory does not consider the constrains related to the *Driver*, the *Vehicle* and the *Environment*, a large section is dedicated to the tuning of the Longitudinal Controller. The latter, in a first step, determines a limit speed profile from the generated unconstrained trajectories. This speed profile is then modified in real-time by the Finite State Machine-based controller regarding constraints related either to the *Driver*, the *Vehicle* or the *Environment*.

- The analysis of the considered constraints shows that they are mostly related to the geometric location of the $PCS$ and to its curvature. These constraints can consequently be related to the mathematical expressions of the $PCS$. Then, the formulation of the trajectory generation as an optimization problem helps to integrate these constraints while minimizing a cost criterion which can be the trajectory length, its error compared to a reference trajectory, its strain energy, etc. Finally, the used Lateral Control approach is introduced.

## 3.2 Problem Statement

Consider, as presented in Fig.3.1, a *Vehicle* driving in its driving lane. This situation is subject to constraints (cf. Section.1.4.1). Indeed, normal driving conditions imply that the *Vehicle* stays in its current lane. For Wheeled Rolling Systems ($WRS$) such as car-like vehicles, this area is defined by the road lane on which the *Vehicle* is located. Considering a driving lane of width $L$, the reachable driving area named here *Validity Area*, taking account of the absolute and relative inaccuracies $\epsilon_w$ of the digital map, has a width $VAW = L - (W + \epsilon_w)$ with $W$ the *Vehicle* width. Note that, in the proposed approach, the road is considered to be flat, consequently only 2-dimensional parametric paths $(x(t),\ y(t)) \in \mathbb{R}^2$ are computed with $t \in \mathbb{R}^+$ the parameter . In addition, it is also important to remind that vehicles are subject to limitations (cf. Section1.4.1), and that the *Driver*'s safety and comfort is desired to be maintained along the trajectory.

The present objective is then to provide reference trajectories through an unconstrained and constrained trajectory generation approach. If the unconstrained *Reference* approach involves the consideration of the constraints in the *Controller* synthesis step, the constrained trajectory generation is desired to satisfy several limitations linked to the $WRS$ (the *Vehicle*), the road to be followed (the *Environment*) and finally the *Driver* (the *Controller*), by considering control-oriented constraints during the generation. Giving a starting configuration $q_0 = (x_0, y_0, \psi_0, \kappa_0, \dot{\kappa}_0) \in \mathbb{R}^5$ defined by the $WRS$'s Centre of Gravity ($CoG$) position $(x_0,\ y_0)$, the $WRS$'s orientation $(\theta_0)$, an initial curvature $\kappa_0$ of the path to be followed by the $CoG$, its respective derivative $\dot{\kappa}_0$ and a final configuration $q_n = (x_n, y_n, \psi_n, \kappa_n, \dot{\kappa}_n)$, a continuous sequence of reachable configurations $(x, y, \theta, \kappa, \dot{\kappa})$ defined with respect to the constraints to be verified has to be generated. This set of reachable configurations suits the limitations of the controlled system. In that case, the repartition of the constraints is simultaneously done over the *Reference* and over the *Controller*.

If these trajectory generation solutions have different approaches and different results, they both need to know the location of the driving lane. Indeed, the unconstrained trajectory generation defines the best location for the shape points to be interpolated in the current driving lane, and more precisely in the center of the driving lane. In addition, the constrained trajectory generation requires the definition of a validity area which is also located in the current driving lane. However, the used *Digital Map Database* only gives points which are related to the center of the road. This problem can be overcome by using a road model calculation algorithm.

## 3.3 Road Model Generation

Here, a two step road model calculation is proposed:

- The first step consists in the definition of the relevant shape points of the road as presented in Fig.3.2.b. In other terms, this consists in the definition of the shape points related to the left and right boundaries of the road based on the information provided by the road

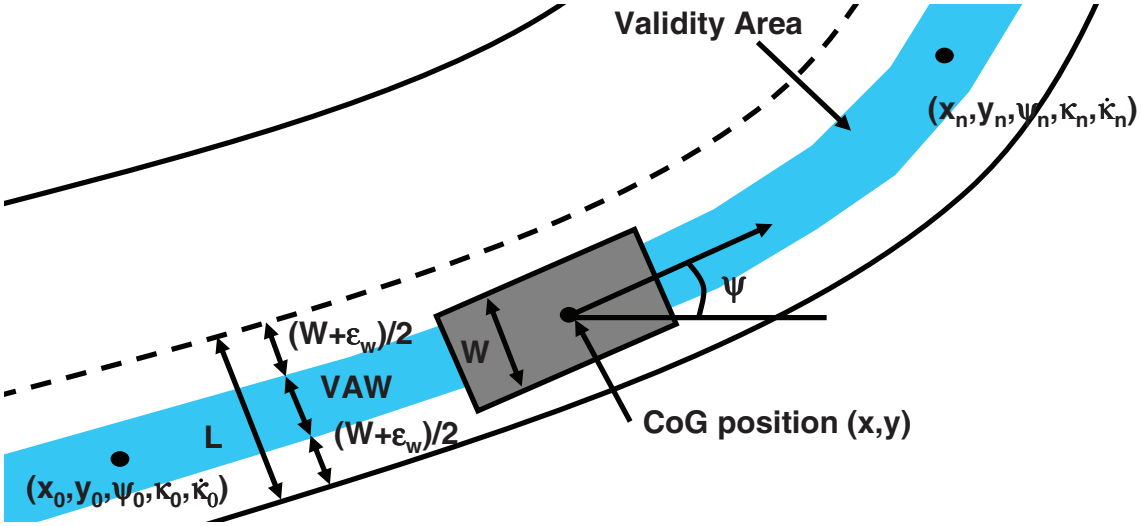



Figure 3.1: Considered Problem

centerline points. By a simple trigonometric relation considering the lane width $L$, the left road boundary points ($LBP_i$) and the right road boundary points ($RBP_i$) can be obtained using (3.1) as described by Fig.3.2.b for the second and third shape points.

$$\begin{aligned} \text{LBP}_\text{i} &\begin{cases} X_{LBP_i} = X_i - L\sin(\alpha_i) \\ Y_{LBP_i} = Y_i + L\cos(\alpha_i) \end{cases} \\ \text{RBP}_\text{i} &\begin{cases} X_{RBP_i} = X_i + L\sin(\alpha_i) \\ Y_{RBP_i} = Y_i - L\cos(\alpha_i) \end{cases} \end{aligned} \tag{3.1}$$

with $X_i$ and $Y_i$ the coordinate of the road centerline points and $\alpha_i$ the different angles.

If this trigonometric translation gives a first set of road boundary shape points, it is still limited. Indeed, for non-extreme points the road boundary shape points can be determined using the current angle $\alpha_i$ but also regarding the previous angle $\alpha_{i-1}$. In the example presented in Fig.3.2.b, this means that the road boundary shape points at the second road point ($LBP_2$ and $RBP_2$) can be determined regarding $\alpha_2$ but also regarding to the previous angle $\alpha_1$ using (3.1). These possible shape points are represented by green points in Fig.3.3.a. The location possibilities for the different road boundary shape points also lie in the different rectangles described by the two green points.

In the following parts of this document the road boundary shape points are considered to be located right in the middle of each solution segments joining the red points in Fig.3.3, so defined by (3.2):

$$\begin{aligned} \text{LBP}_\text{i} &\begin{cases} X_{LBP_i} = \frac{(X_i - L\sin(\alpha_i)) + (X_i - L\sin(\alpha_{i-1}))}{2} \\ Y_{LBP_i} = \frac{(Y_i + L\cos(\alpha_i)) + (Y_i + L\cos(\alpha_{i-1}))}{2} \end{cases} \\ \text{RBP}_\text{i} &\begin{cases} X_{RBP_i} = \frac{(X_i + L\sin(\alpha_i)) + (X_i + L\sin(\alpha_{i-1}))}{2} \\ Y_{RBP_i} = \frac{(Y_i - L\cos(\alpha_i)) + (Y_i - L\cos(\alpha_{i-1}))}{2} \end{cases} \end{aligned} \tag{3.2}$$

- The second step is dedicated to the determination of a continuous expression for the road boundaries. To do this, the boundary shape points determined previously are interpolated

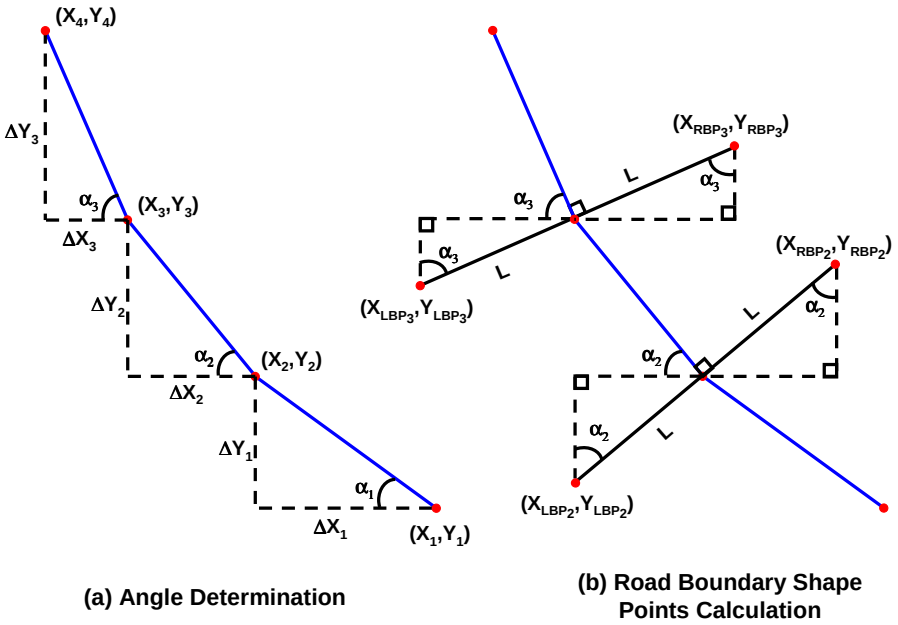



Figure 3.2: Road Points Translation

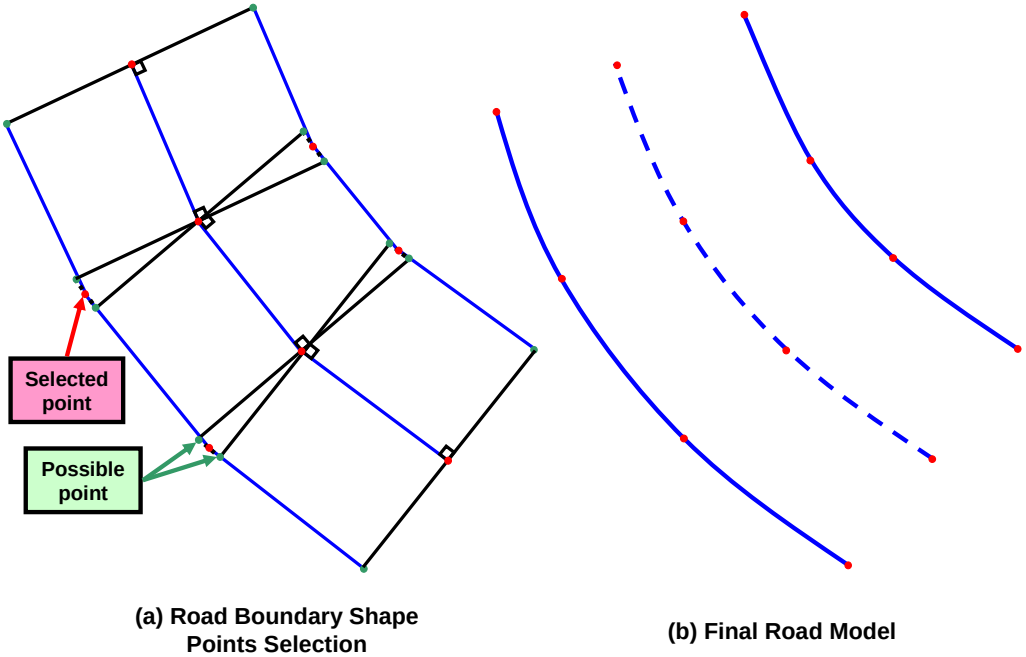



Figure 3.3: Road Model Calculation

via the *PCS* interpolation. This helps to define smooth and curvature-continuous boundaries as presented in Fig.3.3.b, so describes a road model which is closer to the shape of the real road. Furthermore, using Spline to calculate the road model is a quite accurate solution as presented in Section.3.4.1.3 and in Section.5.2. In these sections, it will be shown that the road model is very close to the reality. Finally, the Spline road model is essential to the constrained trajectory generation as it helps to express the constraints with Spline coefficients-related relations (cf. Section.3.5.1).

## 3.4 Unconstrained Trajectory Generation

The unconstrained approach helps to show the benefits of the selected mathematical model - Parametric Cubic Splines -, i.e. to show the results of the tracking problem when all the constraints are devoted to the *Controller*, and finally to provide a *Reference* for the comparison with the constrained trajectory generation. The strategy introduced in Chapter.2, explicitly shows that the unconstrained trajectory generation is based on the determination of shape points which are then interpolated by a *PCS*. This section aims at describing in detail the solution that has been retained and finally tested on a *Longitudinal Controller*.

### 3.4.1 Splines as a Trajectory Generation Solution

#### 3.4.1.1 Interpolation Conditions

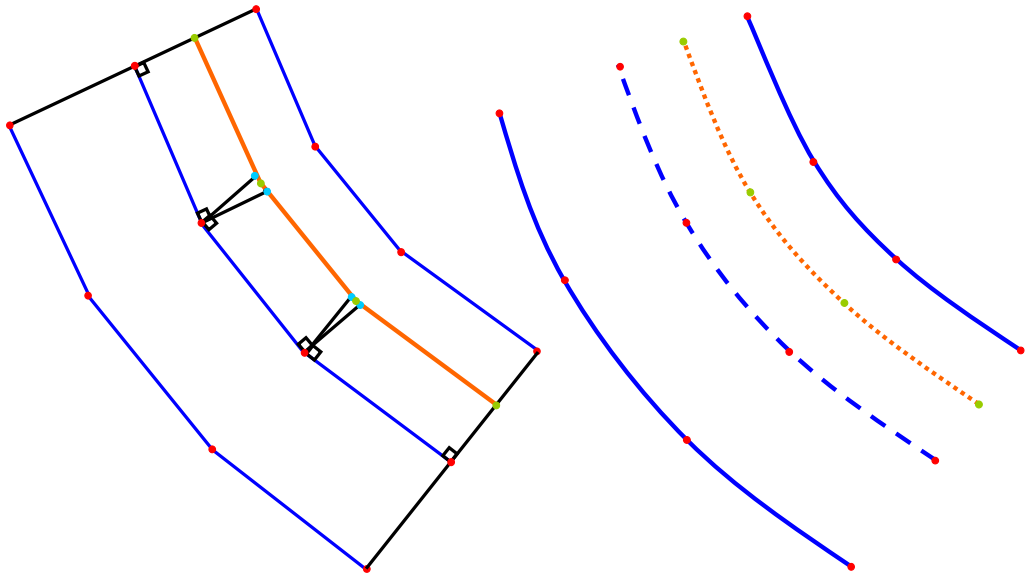
This generation algorithm is strongly dependent on the location of the shape points to be interpolated as they define the set of locations which have to be included in the trajectory. Even if the unconstrained trajectory cannot include the control-oriented constraints related to the current specifications, it has still to grant the essential aspect related to the fact that the *Vehicle* has to stay in the validity area. The shape points to be interpolated have consequently to be located in the current driving lane. In addition, the unconstrained trajectory is, as the constrained trajectory, desired to suit a large panel of controllers requirements. Usually, in the Lateral control domain, the normal trajectory is said to be located right in the middle of the driving lane without performing any special maneuvers [Minoiu Enache et al., 2009]. This approach has been retained for the proposed unconstrained trajectory generation.

To generate the centerlane points, a strategy similar to the road model calculation has been retained: a trigonometric translation of the road centerline points. The difference lies in the fact that only one additional set of points is generated such as presented in Fig.3.4.a.

These points are then processed through the *PCS* interpolation, so providing a smooth curvature continuous curve as presented in Fig.3.4.b. As mentioned earlier, a parametric Spline is a piecewise polynomial interpolation which provides first and second derivative continuities at interpolated points, so provides curvature continuity along the trajectory. It has also been stated that an infinity of Splines are available for one set of shape points. These Splines are differentiated by two tuning parameters: boundary conditions and parameter values which have a large impact on the Spline shape. The determination of these tuning parameters are detailed in the two following sections.

#### 3.4.1.2 Choosing Continuity Conditions

The Spline curvature continuity is provided by the definition of first and second derivative continuity conditions on the interpolated points. These continuities are provided by the Spline



(a) Lane Shape Points Calculation and Selection

(b) In-lane Trajectory

Figure 3.4: Centerlane Trajectory Determination

interpolation algorithm for each point except the extreme points (the first and last points). The determination of the first and second derivative values at these particular locations is known as the Spline boundary conditions definition. Boundary conditions selection can be done using numerous solutions: the conditions can be equal to zero (natural conditions), arbitrarily defined (generalized conditions), etc. [Boor, 1978]. To select the appropriate solution, different boundary conditions have been tested and compared to real road data. Results of this study are depicted in Fig.3.5.

Here the goal is to compare Splines to the road centerline. The road data (in full and dashed black) has been provided by the road builder, and corresponds to real measurements. Note that here, the road corresponds to the last section of a race circuit which has a $9m$ width. Using the road centerline shape points (red points), three different Splines have been generated. The first one (in dash/dot red), is obtained using natural conditions, the second one (in dashed blue), is obtained using generalized conditions and the last one (in full pink) is obtained using special conditions that have been specifically developed and empirically defined for car-like vehicles trajectory generation. These are of the following form:

$$\ddot{f}(t_0) = \begin{cases} 2\frac{\left(\frac{X_2-X_1}{h_1}\right)-\left(\frac{X_1-X_0}{h_0}\right)}{h_1+h_0} \\ 2\frac{\left(\frac{Y_2-Y_1}{h_1}\right)-\left(\frac{Y_1-Y_0}{h_0}\right)}{h_1+h_0} \end{cases}$$
$$\ddot{f}(t_n) = \begin{cases} 2\frac{\left(\frac{X_n-X_{n-1}}{h_{n-1}}\right)-\left(\frac{X_{n-1}-X_{n-2}}{h_{n-2}}\right)}{h_{n-1}+h_{n-2}} \\ 2\frac{\left(\frac{Y_n-Y_{n-1}}{h_{n-1}}\right)-\left(\frac{Y_{n-1}-Y_{n-2}}{h_{n-2}}\right)}{h_{n-1}+h_{n-2}} \end{cases} \tag{3.3}$$

with $h_i = t_{i+1} - t_i$.

As they are only used for the definition of the first point and the last point second deriva-

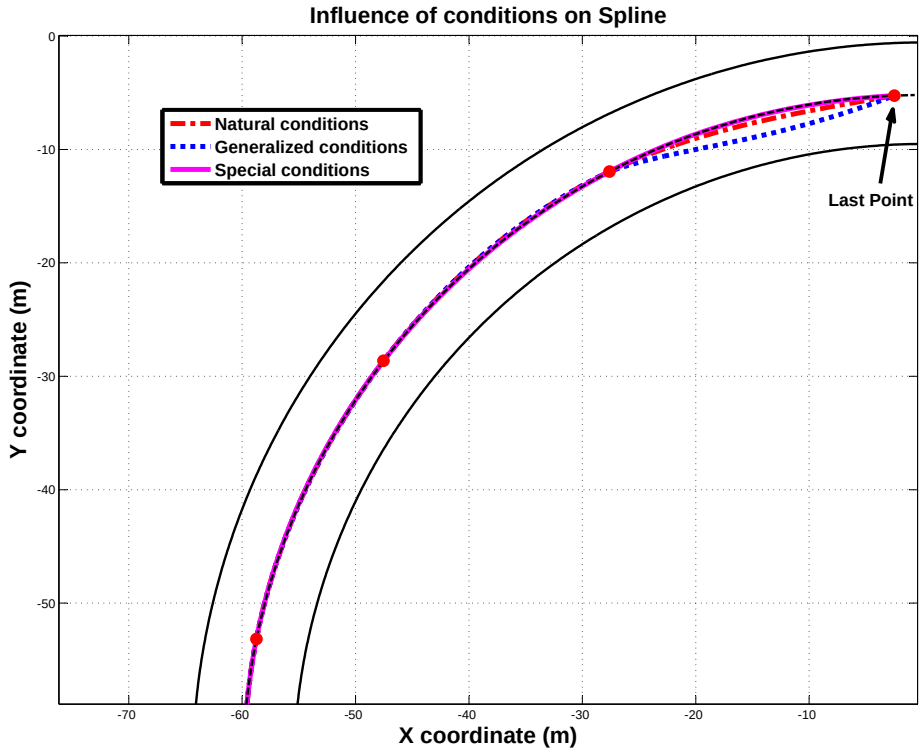



Figure 3.5: Influence of conditions on Spline

Table 3.1: Influence of Conditions on Spline

| *Conditions* | *Natural* | *Generalized* | *Special* |
|---|---|---|---|
| *Maximum Difference (m)* | 0.58 | 1.81 | 0.56 |
| *Mean Difference (m)* | 0.14 | 0.21 | 0.10 |

tives conditions, they are only acting on the corresponding two Spline polynomials which are respectively after and before the first and last point. In Fig.3.5, the end of a Spline defined regarding these three methods are presented. Before the two last polynomials, the Splines are overlapped. However, the figure shows that the Spline defined with special conditions is matching the real road data with a better precision in the two last intervals. This is confirmed by the numerical results presented in Table.3.1. The latter describes the results obtained after studying the difference between the road centerline and the different Splines for the portion of the test track presented in Fig.3.5. As visible in the figure, the Spline generated with special conditions presents the best results. This solution has consequently been retained for the unconstrained trajectory generation approach.

#### 3.4.1.3 Choosing Parameter Expression

The second element which can act on the Spline shape is the definition of the parameter $t$. As mentioned previously, parametrization gives more freedom to the Spline, but it also adds another variable in the computation process which values $t_i$ have to be carefully chosen. Well known techniques to define Spline parameter values are based on ([Floater, 2008]):

- Linear relation:

Table 3.2: Influence of Conditions on Spline

| *Parameter Values* | *Linear* | *Chordal* | *Centripetal* |
|---|---|---|---|
| *Maximum Difference (m)* | 0.63 | 1.15 | 0.56 |
| *Mean Difference (m)* | 0.26 | 0.34 | 0.24 |

$$t_i = i \; with \; i \in [0, n-1] \in \mathbb{N} \tag{3.4}$$

- Chordal relation (distance dependent):

$$t_{i+1} = \sum_{i=0}^{n-2} \left( (X_{i+1} - X_i)^2 + (Y_{i+1} - Y_i)^2 \right)^{\frac{1}{2}} \; with \; t_0 = 0 \tag{3.5}$$

- Centripetal relation (square root of the distance dependent):

$$t_{i+1} = \sum_{i=0}^{n-2} \left( (X_{i+1} - X_i)^2 + (Y_{i+1} - Y_i)^2 \right)^{\frac{1}{4}} \; with \; t_0 = 0 \tag{3.6}$$

Tests on these parameter definitions have been made on the same test site that has been used for conditions determination. Results presented in Fig.3.6 show that the use of centripetal values provides a trajectory that best matches the real road data. Indeed, linear parameter values result in a Spline that presents bad results in bend and chordal values are not providing sufficient efficiency in straight lines. The selection of the parameter values to be used is consequently related to the considered road. Indeed, for linear roads, linear values are preferred to chordal ones, while chordal values are preferred to linear values in circular roads. As roads may have different compositions, a compromise is then to select the centripetal values which provides good results for different road compositions.

As for conditions, an error study of these three different techniques has been carried out. Results of the portion considered in Fig.3.6 are depicted in Table.3.2. This table confirms the performance of the Spline which uses centripetal values. The difference with the real road data is indeed greatly reduced compared to chordal values. However, the improvement compared with the Spline defined with linear parameter values on this road is not evident. Nevertheless, the use of centripetal values is considered in the unconstrained trajectory generation approach.

### 3.4.2 Trajectory Generation for Longitudinal Control

#### 3.4.2.1 Control Strategy

The global structure of the unconstrained trajectory generation for Longitudinal Control is presented in Fig.3.7. For clarity reasons, this structure focuses on the strategy adopted for the controller. It can be seen in the figure that the Longitudinal Controller requires a reference speed $V_{ref}$ which is provided by a *Reference Generator*. The latter is based on the unconstrained trajectory and on a speed profile model to define the suited reference speed. This speed is then compared to the *Vehicle* speed $v$ and the error to the reference $\epsilon_v$ is used by a Finite State Machine ($FSM$) to determine the current state $q$ (*Accelerating*, *Maintaining*, *Braking*) of the *Vehicle* [Daniel et al., 2009a]. Then considering these two elements, control signals $u$ are generated (cf. Section.3.4.2.5). A feedback of the position and speed to the Controller and to the *Reference Generator* is then performed.

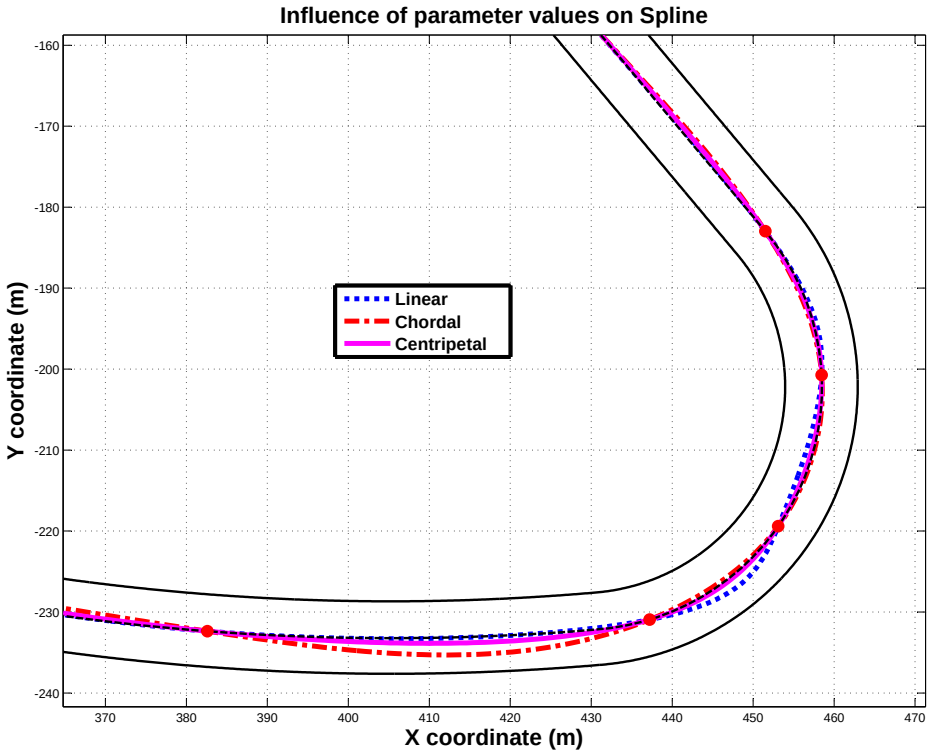



Figure 3.6: Influence of Parameter Values on Spline

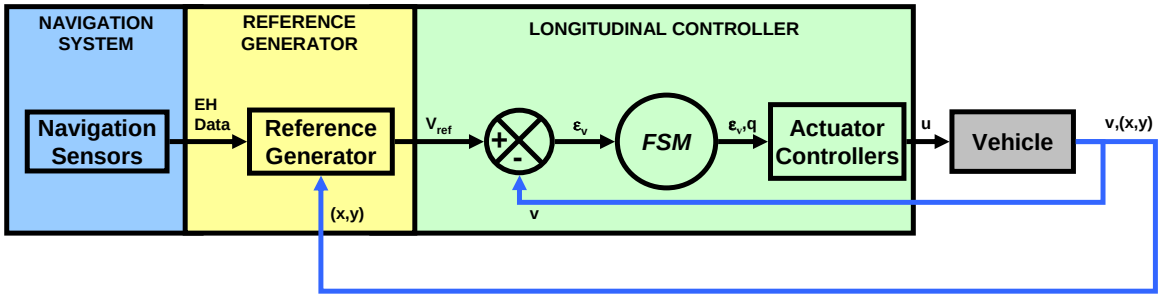



Figure 3.7: Longitudinal Control Strategy

#### 3.4.2.2 Speed Profile Generation

As mentioned previously, the $PCS$ model helps to define smooth and curvature continuous curves which may be used by different control-oriented $ADAS$. The *Reference Generator*, which purpose is to define the reference speed to be sent to the $FSM$, is based on the information provided by the unconstrained trajectories. These trajectories are used to define a continuous limit speed profile, which corresponds to the maximum speed the *Vehicle* can take to cover the considered road section.

In the literature, several methods help to determine the limit speed profile of a trajectory using different speed models (cf. [Glaser et al., 2007]). Here, a straightforward model is considered (cf. (3.7)) as:

- The elevation of the road is not taken into account in the chosen model. This information which is not included in the *Digital Map Database* has a slight impact on the reference speed,
- The road friction coefficient which characterizes the contact between the tires and the road is not involved in the reference speed model. The main reason for this choice is the difficulty to obtain this physical quantity in real time with economically acceptable solutions,
- Considering the fact that the proposed system is an assistance system, the general *Environment* of the *Vehicle* is not taken in account: traffic signs, other vehicles, etc, are not involved in the definition of the speed model. The *Driver* determines himself if the road conditions are adequate or not (limited traffic, good road visibility, etc.) and can control the system insofar as this assistance system can easily be switched off. This is necessary when another *Vehicle* appears or when there is a change in the traffic situation. Further improvements would be focused on the integration of the *Vehicle Environment* in the control system.

The *Vehicle* acceleration, on a planar road, is always composed of two elements: the longitudinal and the lateral acceleration. The accelerations are used by the *Driver* to evaluate his safety and comfort feeling. Considering bending conditions, it is commonly established that the lateral acceleration is bounded by a typical maximum value of $3m.s^{-2}$ [ISO, 1997]. The second parameter of the selected reference speed model is the curvature which is extracted from the generated trajectory. The maximum reference speed is consequently computed as follows:

$$V_{limit} = \sqrt{\frac{\gamma_{T_{max}}}{\kappa}} \tag{3.7}$$

with $\gamma_{T_{max}}$ the maximal allowable lateral acceleration and $\kappa$ the road curvature. As the lateral acceleration limitation is a fixed parameter in the reference speed model, only the curvature determines the limit speed profile. The curvature $\kappa$ is extracted for each trajectory points using the classical formula of parametric curve curvature expressed as:

$$\kappa\left(x(t), y(t), t\right) = \frac{\ddot{y}\left(t\right)\dot{x}\left(t\right) - \ddot{x}\left(t\right)\dot{y}(t)}{\left(\dot{x}^2\left(t\right) + \dot{y}^2\left(t\right)\right)^{\frac{3}{2}}} \tag{3.8}$$

with $x(t)$ and $y(t)$ the two cubic Splines of the $PCS$. Results of the limit speed profile generation are presented in Section.5.3.

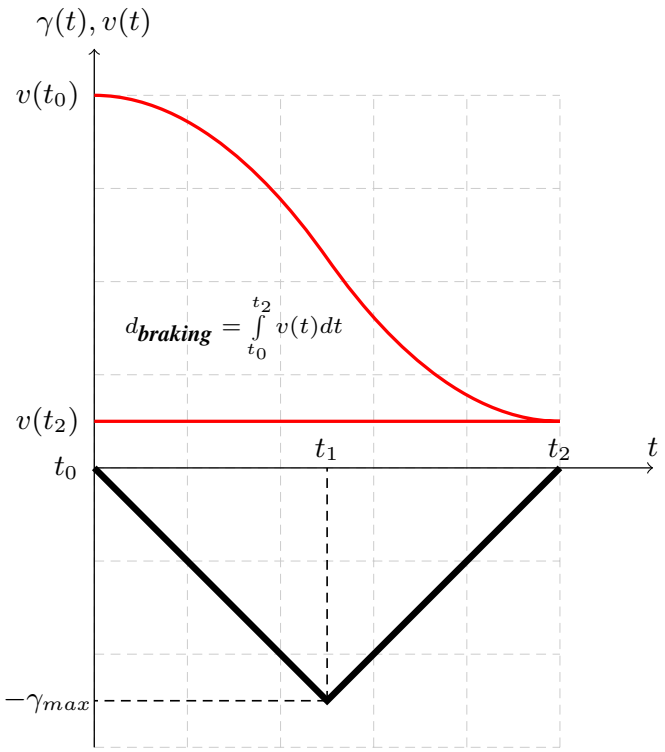



Figure 3.8: Deceleration (bottom) and Speed Profile (top)

#### 3.4.2.3 Adapting the Speed Reference

As currently defined, the limit speed profile cannot be directly used since it does not take account of the *Driver/Vehicle/Environment* constraints. Indeed, it is only based on the geometric properties of the trajectories and on the *Driver* comfort through the consideration of an instantaneous maximal lateral acceleration of $3m.s^{-2}$. However, it is not considering the prediction capabilities of the *Driver*. The latter is aware of acceleration and braking capabilities of his *Vehicle*. He is consequently trying to define the different actions to be done for a safe driving with a certain prediction level. For example when a bend is detected, he is planning a smooth deceleration phase which grants his feelings and mostly his comfort. This avoids hard braking phases which involve stressful, so dangerous situations. Moreover, this model does not consider the *Vehicle* deceleration capabilities and depending on the situation, the required control signal may exceed the maximal achievable braking pressure.

To reproduce the *Driver*'s behavior and avoid excessive braking, the present solution aims at taking account of these considerations at the controller level. The limit speed profile is then combined with a deceleration profile to improve safety and comfort. This solution helps to limit the maximum deceleration, so that the *Vehicle*'s actuators can follow the calculated control signal. Moreover, this deceleration profile allows the definition of the deceleration distance. The chosen deceleration profile is close to the deceleration performed by a common *Driver* and reproduces a smooth braking behavior. The deceleration gradually increases up to a maximum and then decreases back to zero, giving the triangular deceleration represented in Fig.3.8.

Based on this deceleration profile and considering the positioning and speed of the *Vehicle*, it is possible to determine if the previously defined reference speeds ahead of the *Vehicle* are reachable or not. To test the reachability of the limit speed values, the computation of the

braking distance ($d_{braking}$) is done:

$$d_{braking} = \frac{3}{\gamma_{T_{max}}} (\Delta V)^2 + 2(2V_{current} - V_{new}) \frac{\Delta V}{\gamma_{T_{max}}} \tag{3.9}$$

Considering $\Delta V = V_{new} - V_{current}$ with $V_{new}$ the speed to be reached, the maximum allowable deceleration of $\gamma_{T_{max}} = 3m.s^{-2}$ and $V_{current}$ the current *Vehicle* speed, the braking distance $d_{braking}$ helps to calculate the distance necessary to reach the speed defined by the limit speed profile. If the different speeds are reachable, in other words, if the distance between the *Vehicle* and the point corresponding to $V_{new}$ is larger than $d_{braking}$, the *Vehicle* continues to follow the initial speed reference. If not, the *Vehicle* decelerates following the deceleration profile, so that:

$$\begin{aligned} V_{ref} &= \frac{\gamma_{T_{max}}^2}{2(V_{new} - V_{current})} t^2 + V_{current} && t \in [t_0, t_1] \\ V_{ref} &= \frac{\gamma_{T_{max}}^2}{2(V_{new} - V_{current})} t^2 + 2\gamma_{T_{max}} t + 2V_{current} - V_{new} && t \in [t_1, t_2] \end{aligned} \tag{3.10}$$

In summary, this speed profile which integrates a classical *Driver* behavior and the *Vehicle* limitations as constraints is safer and more realistic than the limit speed profile directly obtained from the unconstrained path. Finally, an additional constraint has been added. The latter aims at reproducing the classical behavior of the *Driver* who defines the set of actions to be done for a safe driving by anticipation. Concerning the longitudinal control, he evaluates the convenient speed to cover the considered road section also by anticipation. The same strategy has been implemented in the Longitudinal Controller with the use of a Look Ahead Distance ($LAD$). This distance helps to consider a reference speed at a further speed profile point, thus introducing the *Driver* foreseeing ability. Drivers usually need 1 second to react. With an additional safety margin of 1 second, the control system is also considering a reference speed located at 2 second further on the road. The $LAD$ is also defined as: $LAD = 2.V(t)$. The Longitudinal Controller consequently uses an improved definition of the error $\epsilon$ such that:

$$\epsilon_v = V_{ref}(X_{LAD}, Y_{LAD}) - V(t) \tag{3.11}$$

with $V_{ref}(X_{LAD}, Y_{LAD})$ the reference speed of a trajectory point located ahead of the *Vehicle* at a distance $d_{LAD}$. In other words, the $(X_{LAD}, Y_{LAD})(t)$ trajectory point is defined by solving

$$d_{LAD} = \sqrt{(x(t) - X_{LAD}(t))^2 + (y(t) - Y_{LAD}(t))^2} \tag{3.12}$$

Using this $LAD$, the system reacts in advance, as the *Driver* does, and the control is less affected by the *Vehicle* response time.

In conclusion the reference speed used by the control loop, $V_{ref}$, considers several constraints (the geometrical aspects of the trajectory, the *Driver* comfort through the limited lateral acceleration) and reproduces the *Driver*'s behavior through the calculation of an improved speed limit and through the determination of the reference speed via the $LAD$.

#### 3.4.2.4 Speed Controller

The control solution is based on a Finite State Machine ($FSM$). This is a triplet $(Q, \Sigma, \mathcal{T})$ such that $Q$ is a finite set of states, $\Sigma$ the finite state of events and $\mathcal{T}$ the transition function which specifies the set of possible states following a particular event. In the present application, the $FSM$ is defined with $Q = \{$*Accelerating, Maintaining, Braking*$\}$ where:

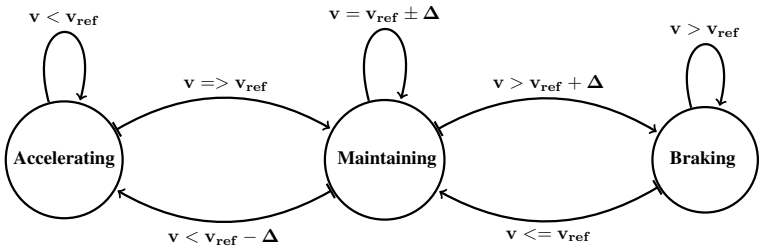



Figure 3.9: The FSM with the Tolerance

- *Accelerating* occurs when an increase of speed is requested, this may correspond to a situation in which the *Driver* leaves a bend or when the current speed does not provide *Driver*'s satisfaction.
- *Maintaining* occurs when the *Vehicle* speed corresponds to the limitation or when it fulfills the *Driver*'s satisfaction.
- *Braking* occurs when an obstacle appears, when the speed limitation decreases or when the *Driver* negotiates a bend.

The fact that speed sensors give imprecise and inaccurate measurements must be taken into account. Consequently, the $FSM$ which uses these measurements may be subject to state oscillations. To cope with the problem, the $FSM$ and particularly the transition conditions of the *Maintaining* state have been improved. A tolerance speed $\Delta$ has been added, so that, for example, the transition conditions between the *Maintaining* state and the *Accelerating* state becomes $V < V_{ref} - \Delta$. The $FSM$ is shown in Fig.3.9.

This improved $FSM$ helps to limit the state changes and so provides smoother feeling. It also corresponds to the way the *Driver* behaves as he cannot control his speed with a high precision. The choice of the tolerance speed was determined experimentally and is a compromise between *Vehicle* passenger comfort and acceptance of the delay induced. It is clear that the addition of this contribution delays the transition of the $FSM$. Considering, for example, the braking conditions, an important tolerance speed postpones the braking action and may lead to hazardous situations. However, the use of the $LAD$ to determine the reference speed diminishes the delay induced in the tolerance speed.

#### 3.4.2.5 Actuators

The actuators considered here are the Braking System ($BS$) and the Cruise Control system ($CC$) of the test car (cf. Section.A.2). The $BS$ is characterized by an input/output transfer function (voltage/speed). This system is controlled with a Proportional Integral Derivative ($PID$) feedback controller. It provides good control performance despite the variation of the process dynamic and is obtained by the use of standard tuning rules (Ziegler-Nichols, Cohen-Coon and direct synthesis).

The $CC$ system has three digital control signals (cf. Section.A.2) which are: $ACC$, $DCC$ and $Cancel$. These are directly acting on the $CC$ internal control loop. The Longitudinal Control is done by the $FSM$ and by an appropriate selection of the aforementioned digital signals.

For each active $FSM$ state, a set of actions for the $CC$ and for the $BS$ are triggered which have a direct impact on the *Vehicle*.

### 3.4.3 Control-oriented Constraints Management

From this Longitudinal Controller presentation, it has been clearly shown that all the *System*'s constraints are focused on the *Controller*. The proposed approach deals with several types of constraints/limitations due to the *Driver*, the *Vehicle*, the sensors or even the *Environment*. However, it also proves the importance of the *Reference* as the speed reference, determined by the *Reference Generator* is still the central element for the Longitudinal Controller. Indeed, it represents the safe speed regarding to the road geometry which has to be reached by the *Vehicle*. Moreover, even if the unconstrained trajectory generation does not explicitly integrate control-oriented constraints, it implicitly considers almost all the constraints related to the road geometry (in other words to *Environment* constraints): it keeps the *Vehicle* in its driving lane (cf. Section.3.2). This is confirmed by the results presented in Section.5.3.

## 3.5 Constrained Trajectory Generation

### 3.5.1 Multiple Constrained Path Generation

Cars are known to be non-holonomic systems: not all the solutions of the configuration space are possible and limitations in the directions of motion have to be taken into account (cf. [Fraichard and Scheuer, 2004]). An effective path generation algorithm should consider these limitations when computing the nominal trajectory the *Vehicle* must follow.

This study considers three types of constraints. The first ones are geometric constraints linked, on the one hand, to the configuration space in which the *Vehicle* moves (to keep the *Vehicle* in a prescribed driving area) and on the other hand, to the mechanical limits of the *Vehicle*. Due to the non-holonomy property of $WRS$, these vehicles are subject to kinematic constraints. This second type of constraints involves (linear and rotational) velocities and acceleration limits of the kinematic variables describing the *Vehicle* (cf. [Graettinger and Krogh, 1989] for a description of these limits in the robotic domain). Finally, the third category of constraints are dynamic limitations which characterize the performance of the *Vehicle* (braking/accelerating capabilities, maximum lateral acceleration, tyre/ground interaction etc.) and the actuators performance. They restrict the overall performance of the *Vehicle*.

It is important to note that the present approach is completely different from the unconstrained trajectory generation presented in the previous section. Indeed, the control-oriented constraints are here directly considered during the *Reference* definition.

#### 3.5.1.1 Geometric Constraints

The geometric constraints are of two types: limitations related to the configuration space and limitations related to the mechanical conception of the $WRS$. The first constraints define the area prescribed in the configuration space, i.e. the set of possible configurations, in which the $WRS$ is allowed to move. This is represented by the validity area in Fig.3.1.

Concerning the mechanical constraints of $WRS$, the limitations of the steering system imply a minimum break radius $R_{min}$ of the *Vehicle*. This turning radius is lower-bounded and consequently, the trajectory instantaneous curve radius must be constantly greater than the lower bound *Vehicle* turning radius:

$$\frac{1}{R_{trajectory}} = \kappa_{trajectory} < \kappa_{max1} = \frac{1}{R_{min}} \tag{3.13}$$

Finally, in order to avoid any steering function discontinuities, and since the steering angle is directly dependent on the instantaneous curvature of the trajectory to be followed, a continuous-curvature trajectory is necessary. So, the generated trajectories must be $C^2$-continuous.

#### 3.5.1.2 Kinematic Constraints

The kinematic constraints depend on the speed profile along the trajectory. The main kinematic constraint to be considered for the *Vehicle* is the limitation of the steering velocity. It is well known that the dynamic behavior of the steering system is upper bounded. Since the steering velocity is related to the derivative of the curvature, this implies that the trajectory instantaneous curvature derivative is upper-bounded:

$$\dot{\kappa}_{trajectory} < \dot{\kappa}_{max} \tag{3.14}$$

#### 3.5.1.3 Dynamic Constraints

These constraints are due to the limited and often nonlinear dynamic behavior of the *Vehicle* and its parameters (bounded acceleration capabilities, variable ground/wheel interaction, etc.). They mainly influence the longitudinal and lateral accelerations and thus the velocities of the *Vehicle* as described in Section.3.5.3. In order to provide safe trajectories, the maximum centrifugal acceleration allowed for curve negotiation is considered here as done for the Longitudinal Control. It has been shown that different models linking the centrifugal acceleration to the trajectory curvature are available. This section focuses on the simplified model implemented for the Longitudinal Controller and given by:

$$\kappa_{max2} = \frac{\gamma_{T_{max}}}{v^2} \tag{3.15}$$

with $\gamma_{T_{max}}$ the maximum allowed lateral acceleration and $v$ the *Vehicle* speed. Note that, the *Driver* is sensible to the accelerations and mainly to the lateral acceleration. Considering a maximum value of acceleration $\gamma_{T_{max}}$, driver-dependent factors are taken in account.

This relation implies that the path curvature must be upper bounded in order to ensure a limited centrifugal acceleration $k_{max2}$:

$$\kappa_{trajectory} < \kappa_{max2} \tag{3.16}$$

#### 3.5.1.4 Note About Time Constraints

If the aforementioned constraints directly refer to the current problem, there is one additional constraint category which has to be considered. In Section.1.4.2.3, constrained trajectory generation were presented as algorithms which require strongly longer calculation times compared to unconstrained solutions. However, as these constrained solutions are designed to assist or replace the *Driver* in its driving task, they consequently have to satisfy real-time conditions. In addition, they have to reproduce the *Driver*'s behavior and need to be generated over several hundred meters. As the calculation time required by an optimization algorithm mostly depends on the length of the considered road portion, a compromise between the prediction ability of the algorithm and the calculation time has to be determined to fulfill the real-time constraint.

#### 3.5.1.5 Constraints Analysis

The aforementioned limitations can be classified into two main categories:

- Constraints related to the Spline coefficients. Indeed, it will be shown in the next section that the geometric constraints which restrict the trajectory location in a validity area can be expressed in function of Spline coefficients,
- Constraints related to the curvature. Car-related geometric, kinematic and dynamic constraints are all referring to a limitation on the curvature or on its derivative, so are constraints of the same type and can be summarized as shown in (3.17).

$$\begin{array}{l} \kappa_{trajectory} < \min\left\{\kappa_{max1}, \kappa_{max2}\right\} \\ \dot{\kappa}_{trajectory} < \dot{\kappa}_{max} \end{array} \tag{3.17}$$

### 3.5.2 Trajectory Generation Formulated as an Optimization Problem

This section describes the integration of the different constraints into the selected mathematical model. Considering that Splines have polynomial expressions, the goal of the present algorithm is to find the optimal coefficients $a_{f_{S_i}}$, $b_{f_{S_i}}$, $c_{f_{S_i}}$ and $d_{f_{S_i}}$ for the constrained trajectory $f_i(t)$ over $n$ considered points:

$$f_i\left(t\right)\left\{\ f_{S_i}\left(t\right) = a_{f_{S_i}}t^3 + b_{f_{S_i}}t^2 + c_{f_{S_i}}t + d_{f_{S_i}}\right. \tag{3.18}$$

with:

$$S_i = \left[x_i, y_i\right]^T, \quad i = 0, 1, ..., n-1 \tag{3.19}$$

and $t \in [t_i, t_{i+1}]$ the parameter.

Based on the results presented in Section.3.4.1.3, linear parameter values are used. This avoids the non-linear expression related to the centripetal expression and still generates good results. The different intervals $[t_i, t_{i+1}]$ consequently corresponds to: $[0,1], [1,2], ...[t_{n-1}, t_n]$ so that $t_{i+1} - t_i = 1, \forall i \leq n-1$. This problem is here considered as referring $n-1$ times to the problem of defining $f_i(t)$ on the $[0,1]$ interval.

The formulation of the trajectory generation as an optimization problem is obtained in three steps which constitutes the following three sections: the definition of the equality constraints, of the inequality constraints and of the selected cost criterion to be minimized.

#### 3.5.2.1 Equality Constraints

The first element to be considered is the curvature continuity constraints, so involving $C^0$, $C^1$ and $C^2$ continuities for the trajectory. As the constrained trajectory is based on the parametric cubic Spline model, these constraints are expressed by continuity of the first and second derivative at each interpolated points. These continuities are obtained via the following relation:

$$\left\{\begin{array}{l} f_{S_i}\left(t_{i+1}\right) = f_{S_{i+1}}\left(t_{i+1}\right) \\ \dot{f}_{S_i}\left(t_{i+1}\right) = \dot{f}_{S_{i+1}}\left(t_{i+1}\right) \\ \ddot{f}_{S_i}\left(t_{i+1}\right) = \ddot{f}_{S_{i+1}}\left(t_{i+1}\right) \end{array}\right. \Rightarrow \left\{\begin{array}{l} a_{f_{S_i}} + b_{f_{S_i}} + c_{f_{S_i}} + d_{f_{S_i}} = d_{f_{S_{i+1}}} \\ 3a_{f_{S_i}} + 2b_{f_{S_i}} + c_{f_{S_i}} = c_{f_{S_{i+1}}} \\ 6a_{f_{S_i}} + 2b_{f_{S_i}} = 2b_{f_{S_{i+1}}} \end{array}\right. \tag{3.20}$$

By considering only these equalities the generated path may result in the curve presented in Fig.3.10. The latter shows a curve which possesses the curvature continuity property. However,

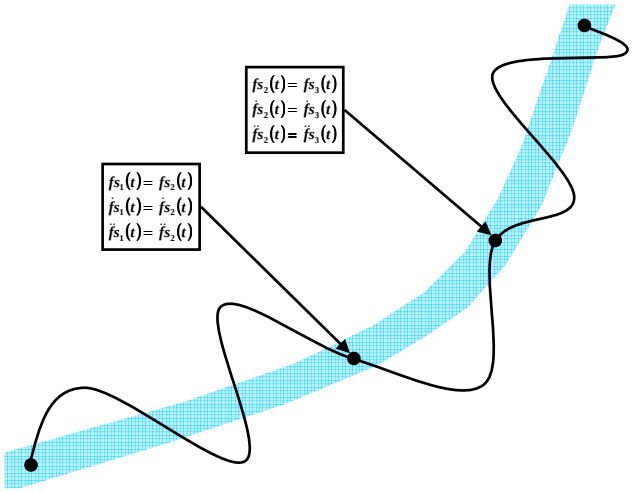



Figure 3.10: Example of Trajectory with Integrated Equalities Only

this curve is fluctuating, in other words present aberrant bends considering the real road context, so may not remain in the limits of the validity area (in shaded blue).

#### 3.5.2.2 Inequality Constraints

The inequality constraints refer to the constriction of the trajectory within the validity area. To do this, let $g_i(t)$ and $e_i(t)$ be the two curves which describe the validity area boundaries. The optimization then consists in finding a Spline $f_i(t)$ such that:

$$e_i\left(t\right) \leq f_i\left(t\right) \leq g_i\left(t\right) \tag{3.21}$$

with:

$$\begin{array}{l} g_i\left(t\right) \left\{ \; g_{S_i}\left(t\right) = a_{g_{S_i}} t^3 + b_{g_{S_i}} t^2 + c_{g_{S_i}} t + d_{g_{S_i}} \right. \\ e_i\left(t\right) \left\{ \; e_{S_i}\left(t\right) = a_{e_{S_i}} t^3 + b_{e_{S_i}} t^2 + c_{e_{S_i}} t + d_{e_{S_i}} \right. \end{array} \tag{3.22}$$

To generate a Spline which stays in the validity area boundaries, the difference between the desired Spline and the upper and lower bounds must be negative or positive respectively:

$$\begin{array}{l} f_i\left(t\right) - g_i\left(t\right) \leq 0 \\ f_i\left(t\right) - e_i\left(t\right) \geq 0 \end{array} \tag{3.23}$$

The question raised by (3.23) more generally concerns Spline positivity conditions regarding its coefficients. The simplest way to ensure the positivity of a cubic polynomial of the form $f\left(t\right) = at^3 + bt^2 + ct + d$ with $t \in [0, 1]$ is when all the coefficients $a$, $b$, $c$ and $d$ are real positive. However, this solution is highly conservative and strongly restricts the set of possible polynomials. In the literature, several studies have been carried out on the positivity of cubic polynomials. Particularly, *Schmidt* showed in [Schmidt and Heß, 1988] that a cubic polynomial $f\left(t\right)$ defined for $t \in [0, 1]$ can be equivalently transformed into:

$$\hat{f}\left(s\right) = \alpha s^3 + \beta s^2 + \gamma s + \delta \tag{3.24}$$

with a variable change $t = \frac{s}{(1+s)}$, $s \in [0; +\infty[$ and with:

$$\begin{cases} \alpha = a + b + c + d \\ \beta = b + 2c + 3d \\ \gamma = c + 3d \\ \delta = d \end{cases} \tag{3.25}$$

The positivity of (3.24) is then defined regarding $\alpha$, $\beta$, $\gamma$ and $\delta$ such as:

$$\begin{array}{l} \hat{f}\,(s) \geq 0 \Rightarrow (\alpha, \beta, \gamma, \delta) \in A \cup B, with \\ \begin{cases} A = \{(\alpha, \beta, \gamma, \delta) : \alpha \geq 0, \beta \geq 0, \gamma \geq 0, \delta \geq 0\} \\ B = \{(\alpha, \beta, \gamma, \delta) : \alpha \geq 0, \delta \geq 0, 4\alpha\gamma^3 + 4\delta\beta^3 + 27\alpha^2\delta^2 - 18\alpha\beta\gamma\delta - \beta^2\gamma^2 \geq 0\} \end{cases} \end{array} \tag{3.26}$$

$A$ and $B$ are the two solution subsets for the different coefficients. The solution subset $A$ presents linear conditions and so, can be easily adapted to the present context. Subset $B$ is defined by a non linear expression involving that the set $A \cup B$ is rather complicated to be used in practice. In the present formulation of the optimization process, the considered subset of possible solutions is limited to $A$.

Based on (3.24), the inequalities of the parameters $a$, $b$, $c$ and $d$ for $f(t)$ are then defined for $t \in [0, 1]$ such that:

$$\begin{array}{l} f\,(t) \geq 0 \Rightarrow (a, b, c, d) \in A, with \\ A = \{(a, b, c, d) : a + b + c + d \geq 0, b + 2c + 3d \geq 0, c + 3d \geq 0, d \geq 0\} \end{array} \tag{3.27}$$

$$\begin{cases} a_{e_{S_i}} + b_{e_{S_i}} + c_{e_{S_i}} + d_{e_{S_i}} \leq a_{f_{S_i}} + b_{f_{S_i}} + c_{f_{S_i}} + d_{f_{S_i}} \leq a_{g_{S_i}} + b_{g_{S_i}} + c_{g_{S_i}} + d_{g_{S_i}} \\ b_{e_{S_i}} + 2c_{e_{S_i}} + 3d_{e_{S_i}} \leq b_{f_{S_i}} + 2c_{f_{S_i}} + 3d_{f_{S_i}} \leq b_{g_{S_i}} + 2c_{g_{S_i}} + 3d_{g_{S_i}} \\ c_{e_{S_i}} + 3d_{e_{S_i}} \leq c_{f_{S_i}} + 3d_{f_{S_i}} \leq c_{g_{S_i}} + 3d_{g_{S_i}} \\ d_{e_{S_i}} \leq d_{f_{S_i}} \leq d_{g_{S_i}} \end{cases} \tag{3.28}$$

The combination of (3.23) and (3.27) leads to the set of linear inequalities presented in (3.28). It can be noticed that the inequalities on $d_{f_{S_i}}$ as they are referring to Spline shape points, could be defined by a rectangle describing the possible solutions for the constrained trajectory shape points as presented in Fig.3.11. In this figure which focuses on a portion of the *Digital Map Database* bend presented in Section.5.4.1.3, an example of constrained trajectory generation is presented. Furthermore, it shows the location of the validity area boundary points (magenta points), the rectangle defined by each pair of boundary points (black rectangle), and the constrained trajectory points (cyan points). In this figure it is clear that the different rectangles define areas which are included in the validity area.

However, in rare cases, rectangles may contain a portion which is located outside of the validity area. Such cases may involve a trajectory which is located outside of the validity area and which is fulfilling the constraints simultaneously. To avoid these situations, another strategy is to consider the rectangle inscribed in the circle defined by each pair of boundary shape points as presented in Fig.3.12.

This technique implies a modification of the inequalities presented in (3.28) such that:

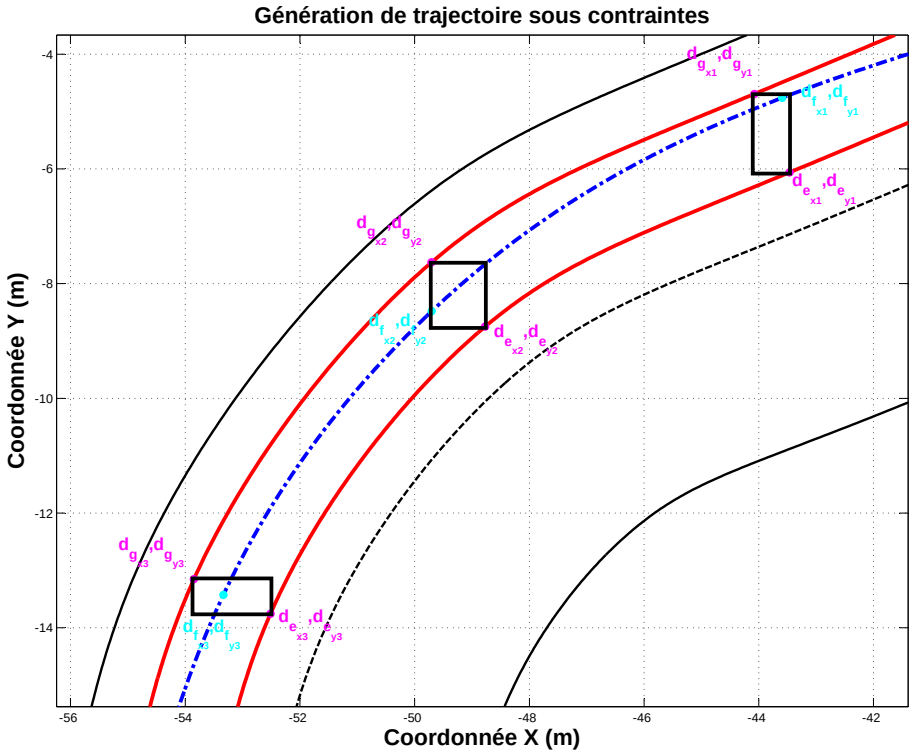



Figure 3.11: Solution Area for the Spline Points

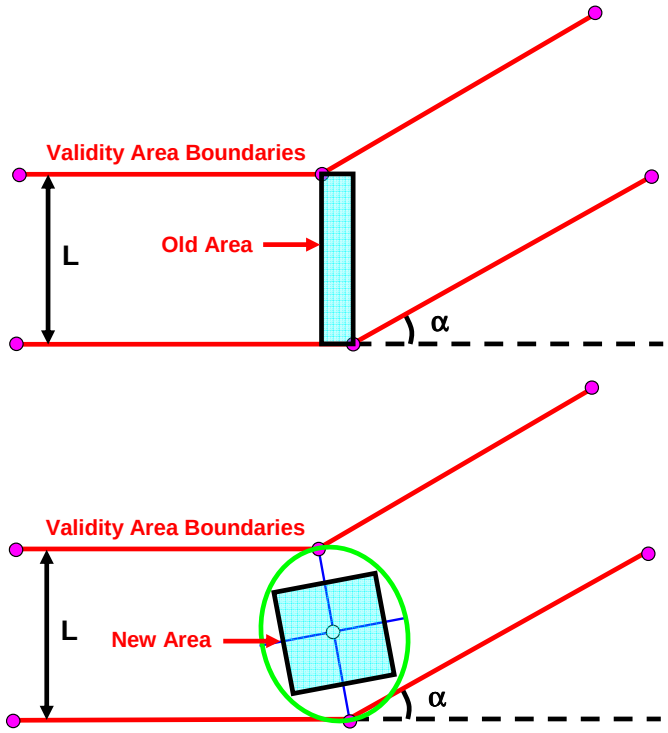



Figure 3.12: Rectangle and Inscribed Rectangle Boundary Areas

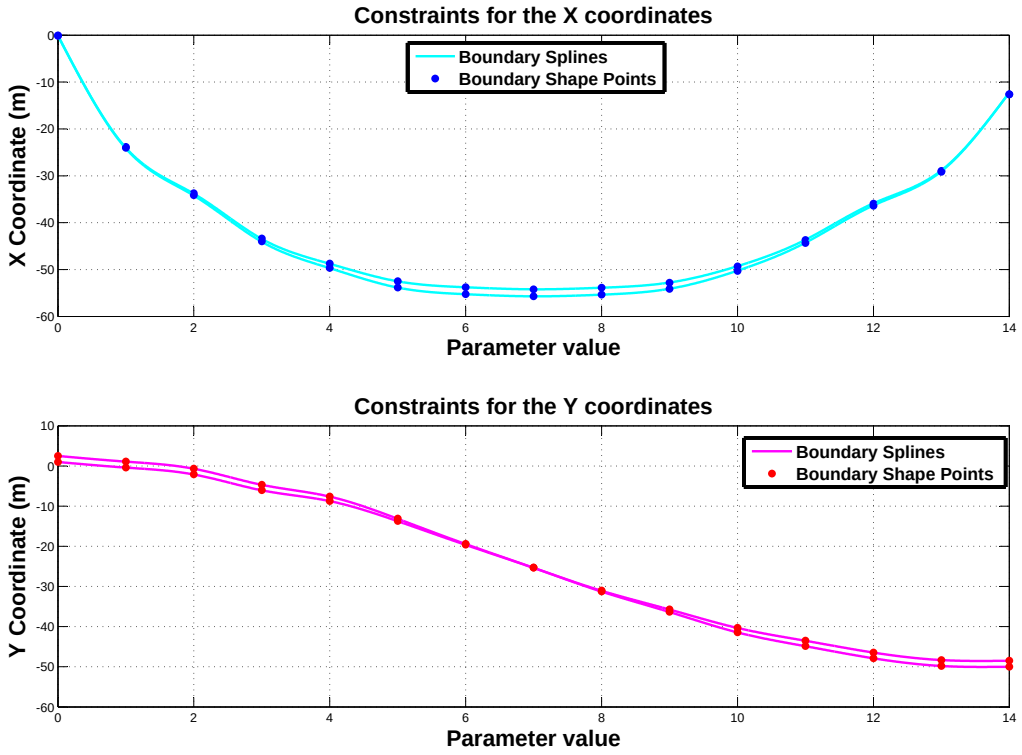



Figure 3.13: Constraints On Coordinates X and Y

$$\begin{cases} a_{e_{x_i}} + b_{e_{x_i}} + c_{e_{x_i}} + d_{e_{x_i}} \leq a_{f_{x_i}} + b_{f_{x_i}} + c_{f_{x_i}} + d_{f_{x_i}} \leq a_{g_{x_i}} + b_{g_{x_i}} + c_{g_{x_i}} + d_{g_{x_i}} \\ b_{e_{x_i}} + 2c_{e_{x_i}} + 3d_{e_{x_i}} \leq b_{f_{x_i}} + 2c_{f_{x_i}} + 3d_{f_{x_i}} \leq b_{g_{x_i}} + 2c_{g_{x_i}} + 3d_{g_{x_i}} \\ c_{e_{x_i}} + 3d_{e_{x_i}} \leq c_{f_{x_i}} + 3d_{f_{x_i}} \leq c_{g_{x_i}} + 3d_{g_{x_i}} \\ -\cos\left(\frac{\pi}{4}+\alpha\right)\left(\frac{d_{g_{x_i}}-d_{e_{x_i}}}{2}\right) + \left(\frac{d_{g_{x_i}}+d_{e_{x_i}}}{2}\right) \leq d_{f_{x_i}} \leq \cos\left(\frac{\pi}{4}+\alpha\right)\left(\frac{d_{g_{x_i}}-d_{e_{x_i}}}{2}\right) + \left(\frac{d_{g_{x_i}}+d_{e_{x_i}}}{2}\right) \\ a_{e_{y_i}} + b_{e_{y_i}} + c_{e_{y_i}} + d_{e_{y_i}} \leq a_{f_{y_i}} + b_{f_{y_i}} + c_{f_{y_i}} + d_{f_{y_i}} \leq a_{g_{y_i}} + b_{g_{y_i}} + c_{g_{y_i}} + d_{g_{y_i}} \\ b_{e_{y_i}} + 2c_{e_{y_i}} + 3d_{e_{y_i}} \leq b_{f_{y_i}} + 2c_{f_{y_i}} + 3d_{f_{y_i}} \leq b_{g_{y_i}} + 2c_{g_{y_i}} + 3d_{g_{y_i}} \\ c_{e_{y_i}} + 3d_{e_{y_i}} \leq c_{f_{y_i}} + 3d_{f_{y_i}} \leq c_{g_{y_i}} + 3d_{g_{y_i}} \\ -\sin\left(\frac{\pi}{4}+\alpha\right)\left(\frac{d_{g_{y_i}}-d_{e_{y_i}}}{2}\right) + \left(\frac{d_{g_{y_i}}+d_{e_{y_i}}}{2}\right) \leq d_{f_{y_i}} \leq \sin\left(\frac{\pi}{4}+\alpha\right)\left(\frac{d_{g_{y_i}}-d_{e_{y_i}}}{2}\right) + \left(\frac{d_{g_{y_i}}+d_{e_{y_i}}}{2}\right) \end{cases} \tag{3.29}$$

Note that the different angles $\alpha$ have been considered here, to take the road direction in account for the definition of the solution area.

If the proposed inequalities avoid situations in which trajectories go out of the validity area, these constraints may lead to bound values which are very close, so that $d_{f_{S_i}}$ can only vary by a few centimeters. The considered inequalities can thus be very conservative as confirmed by Fig.3.13. The latter, corresponds to the graphical representation of the constraints involved by the inequalities on $x = f(t)$ and on $y = f(t)$ over the complete bend described by Fig.3.11. By focusing on the $Y$ coordinate, it can be seen that, at the beginning and at the end of the considered bend ($t \in [1, 5]$ and $t \in [9, 14]$), the validity area is quite large ($1.5m$ of freedom). However, in the center of the bend ($t \in [5, 9]$) the validity area is narrow as the $Y$ coordinate has only a few centimeter freedom ($1.5cm$ at $t = 7$).

The results of this second set of constraints integration is presented are Fig.3.14. The latter

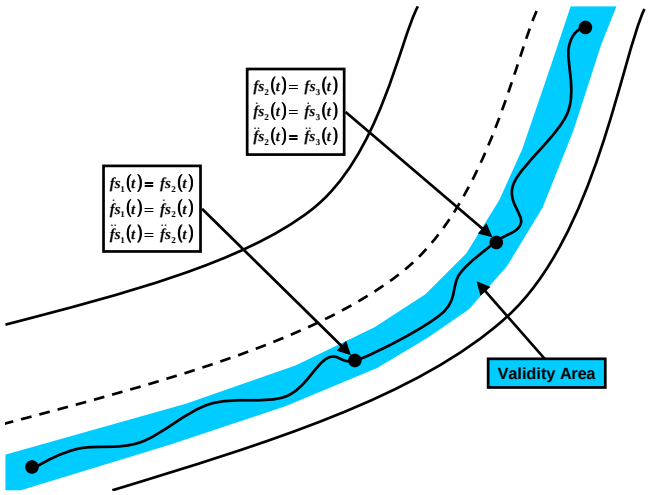



Figure 3.14: Example of Trajectory with Integrated Equalities and Inequalities only

shows a curve which possesses the curvature continuity property and which lies within the validity area boundaries. Nevertheless, this curve is still fluctuating, so may still define undesired and useless maneuvers for a Lateral Controller.

#### 3.5.2.3 Cost Criterion

Most of the optimization algorithms (cf. [Boyd and Vandenberghe, 2004]) allow the minimization of a cost criterion which can be defined regarding various elements or parameters. In the literature, the most common criteria are the minimization of the *Trajectory length*, the minimization of the *Trajectory coverage time* or the minimization of the *Trajectory energy*. Indeed, these cost criteria are in phase with the major requirement of the robotic domain which is power and/or time saving. As the robotic and automotive domains are subject to similar constraints, the same approach can be adopted for the present study. The criteria that are generally considered in optimal trajectory generation are:

- *No criterion.* The solution of the optimization problem then only considers the constraints. In the current context, this type of optimization leads to the generation of a trajectory which only fulfills the geometric constraints related to the validity area boundaries and to the curvature continuity. This optimization is used as a reference for comparison purpose with the other constrained trajectories.

- *Trajectory length.* This common criterion, used in numerous applications, allows the definition of the shortest trajectory which links the current $WRS$ configuration to the desired $WRS$ configuration. In the present context, this corresponds to the definition of the shortest trajectory for the considered road section. For example, it can be straightforwardly described according to the trajectory point coordinates $X_i$ and $Y_i$ which respectively correspond to the $d_{x_i}$ and $d_{y_i}$ Spline coefficients as follows:

$$D = \sum_{i=0}^{n-2} \sqrt{\left(d_{x_{i+1}} - d_{x_i}\right)^2 + \left(d_{y_{i+1}} - d_{y_i}\right)^2} \tag{3.30}$$

- *Distance with a Reference.* This criterion implies the optimization to find the trajectory which is the closest to a pre-defined reference. This is very interesting for specific cases in which some ideal or compulsory trajectory portions are available. This criterion then helps to integrate these locations as constraints in the optimization process and so may provide trajectories of better quality. This solution may also be interesting for the integration of information related to the *Driver*'s experience. In the current study, a particular case has been studied: the reference is described by the unconstrained trajectory defined with linear parameter values (cf. Section.3.4). The expression of this cost criterion is expressed as:

$$DRef = \sum_{i=0}^{n-1} \sqrt{(a_{x_i} - a_{x_{ref_i}})^2 + (b_{x_i} - b_{x_{ref_i}})^2 + (c_{x_i} - c_{x_{ref_i}})^2 + (d_{x_i} - d_{x_{ref_i}})^2}$$

$$+ \sum_{i=0}^{n-1} \sqrt{(a_{y_i} - a_{y_{ref_i}})^2 + (b_{y_i} - b_{y_{ref_i}})^2 + (c_{y_i} - c_{y_{ref_i}})^2 + (d_{y_i} - d_{y_{ref_i}})^2} \tag{3.31}$$

  where $a_{ref_i}$, $b_{ref_i}$, $c_{ref_i}$ and $d_{ref_i}$ are the known coefficients of the reference. It is important to note that the unconstrained Spline can be replaced by reference of different types regarding to the desired application.

- *Trajectory energy.* This criterion refers to the minimization of the trajectory strain energy, which is expressed according to the trajectory curvature $\kappa$ such as mentioned in [Delingette et al., 1991]:

$$E = \int \kappa^2 ds \tag{3.32}$$

  In this particular case, the curvature $\kappa$ corresponds to (3.8). This criterion is in accordance with the constraints of the considered problem which are linked to the trajectory curvature (minimum turn radius, etc.). In addition, energy minimization implies a smoothing of the trajectory curvature which is directly linked to the energy consumption of the *Vehicle*. For example, a trajectory tracking controller, with a smoother curvature, will require less steering energy. Finally, this criterion is well adapted to the current global context of power saving and power consumption reduction.

  By replacing $x(t)$ and $y(t)$ functions into their formal expressions (1.5), (3.8) becomes non-linear (and so becomes (3.32)). The optimal solution (3.32) is also hard to formulate and to compute ([Ye and Qu, 1999]). To overcome this problem, a suboptimal solution is determined. Considering the expression of $x(t) = f_{x_i}(t)$ and $y(t) = f_{y_i}(t)$, the objective is to simultaneously minimize the curvature of each parametric curve $\kappa_x$ and $\kappa_y$ given by:

$$\begin{cases} \kappa_x\left(x\left(t\right),t\right) = \frac{\ddot{x}(t)}{(1+\dot{x}^2(t))^{\frac{3}{2}}} \\ \kappa_y\left(y\left(t\right),t\right) = \frac{\ddot{y}(t)}{(1+\dot{y}^2(t))^{\frac{3}{2}}} \end{cases} \tag{3.33}$$

  Considering that $\dot{x}(t)^2$ and $\dot{y}(t)^2$ are small compared with 1, the energy of the suboptimal solution which corresponds to the cubic interpolation Spline with $C^2$-continuity, can be expressed by:

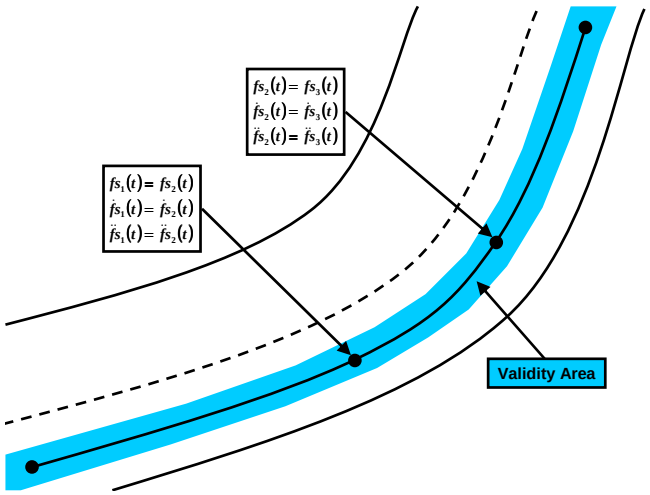



Figure 3.15: Example of Trajectory with Integrated Equalities, Inequalities and Suboptimal Energy Minimization

$$\tilde{E} = \int_{t_0}^{t_n} \left(\ddot{x}^2 + \ddot{y}^2\right) dt \tag{3.34}$$

Regarding the Spline expression (1.5) and its continuity properties, the continuous expression of the energy $\tilde{E}$ can be discretized using the classic Euler method (details are given in Appendix.B). Finally, according to [Pollock, 1999], the total energy is assumed to be of the form:

$$\tilde{E} = \sum_{i=0}^{n-2} \frac{4h_i}{3} \left(b_{f_{x_i}}^2 + b_{f_{x_i}} b_{f_{x_{i+1}}} + b_{f_{x_{i+1}}}^2\right) + \sum_{i=0}^{n-2} \frac{4h_i}{3} \left(b_{f_{y_i}}^2 + b_{f_{y_i}} b_{f_{y_{i+1}}} + b_{f_{y_{i+1}}}^2\right) \tag{3.35}$$

with $h_i = t_{i+1} - t_i$.

These constraints have all been implemented and tested. The results of these tests are available in Section.5.4.

Considering the presented constraints and the strain energy as the criterion will result in a trajectory which is curvature continuous, which remains in the validity area and which minimizes a cost criterion. For the trajectory suboptimal energy, the corresponding path can be represented by the curve in Fig.3.15.

#### 3.5.2.4 Optimization

The trajectory generation formulated as an optimization problem has to fulfill the aforementioned conditions:

- Linear inequality constraints (cf. (3.29)),
- Linear equality constraints (cf. (3.20)),

- A discretized linear or quadratic cost criterion (cf. (3.30), (3.31) and (3.35)).

As mentioned in [Boyd and Vandenberghe, 2004], there are several optimization techniques which help to solve such systems. However, considering the different types of constraints and the fact that they all have a linear or quadratic expression, the convex quadratic programming optimization approach has been chosen:

$$\Delta^* = arg\min_{\Delta} \tfrac{1}{2}\Delta^T H \Delta + f^T \Delta \quad \text{such that:} \begin{cases} A\Delta = B \\ C\Delta \leq D \end{cases} \tag{3.36}$$

with $\Delta \in \mathbb{R}^{[2\cdot4\cdot(n-1)]\times[1]}$ the optimal coefficients of $f_i(t)$ interpolating $n$ points such that:

$$\Delta = [a_{f_{S_1}}, b_{f_{S_1}}, c_{f_{S_1}}, d_{f_{S_1}}, ..., a_{f_{S_{n-1}}}, b_{f_{S_{n-1}}}, c_{f_{S_{n-1}}}, d_{f_{S_{n-1}}}]^T \tag{3.37}$$

and with $A \in \mathbb{R}^{[2\cdot3\cdot(n-2)]\times[2\cdot4\cdot(n-1)]}$, $B \in \mathbb{R}^{[2\cdot4\cdot(n-2)]\times[1]}$, $C \in \mathbb{R}^{[2\cdot4\cdot2\cdot(n-1)]\times[2\cdot4\cdot(n-1)]}$, $D \in \mathbb{R}^{[2\cdot4\cdot2\cdot(n-1)]\times[1]}$.

To solve (3.36), the well known *Dantzig-Wolfe* algorithm [Dantzig and Wolfe, 1960] has been used here. It is clear that this algorithm is well suited for the present problem since:

- The equality and inequality constraints can respectively be written in the $A\Delta = B$ and $C\Delta \leq D$ matrix form,
- The cost criteria, in their discrete formulation, are only expressed using the linear or quadratic terms of the different Spline coefficients (for example $b_{f_{x_i}}$ and $b_{f_{y_i}}$ coefficients for the energy criterion). They can also be easily expressed in the matricial expression $\Delta^T H \Delta$ and $f^T \Delta$,
- This optimization is convex quadratic, so there is only one global minimum,
- This algorithm does not need long calculation time which may coincide with real-time constraints.

#### 3.5.2.5 Remark

It can be noted that geometric constraints (limitations of the configuration space and continuity of the trajectory) are explicitly formulated in the optimization through (3.29) and (3.20) regardless to the cost criterion, contrary to the kinematics and dynamics constraints which are implicitly described by the suboptimal energy cost criteria[18]. Nevertheless, a post-checking of these hypotheses is performed after the optimization. If the checking of the constraints related to the *Vehicle* minimum turn radius and to the *Vehicle* maximum acceleration are straightforwardly determined, note that the checking of the curvature derivative $\dot{\kappa}$ is based on [Fraichard and Scheuer, 2004] which defines $\dot{\kappa}$ for a constant vehicle speed $v$, a wheelbase $b$ and a wheel angle $\phi$ such as:

$$\dot{\kappa} = \frac{\dot{\phi}}{vb\cos^2(\phi)} \tag{3.38}$$

As $\dot{\phi} \leq \frac{\dot{\phi}}{\cos^2(\phi)} \forall \phi$, expression (3.38) can be simplified in a more conservative way by:

$$\dot{\kappa} = \frac{\dot{\phi}}{vb} \tag{3.39}$$

[18] The other criteria are mainly used for comparison purpose.

### 3.5.3 Constrained Trajectory Generation for Lateral Control

To validate the constrained trajectory generation algorithm, it has been applied on a Lateral Controller. As mentioned earlier, the choice of a constrained trajectory implies the constraints to be distributed over the *Reference* and the *Controller*. The section aims at describing the *Trajectory Generation/Controller* system which has been implemented and simulated in the Matlab/Simulink environment.

#### 3.5.3.1 Control Strategy

The type of control which has been used for the Lateral Controller is the Model Predictive Control ($MPC$) developed in [Pouly, 2009]. Two main arguments may explain this choice. On the one hand, the different steps that constitute an $MPC$ are similar to a human decision taking. When a human takes a decision, he observes his *Environment* to define an action and its effect. On the other hand, this control strategy gives the possibility to handle additional constraints and non-linearities, which are not yet considered in the *Reference*, in a rigorous way. Despite some calculation efforts, the non-linearities of the *System* are taken into account insofar as the description of the system evolution is used to obtain the optimal control input.

The general principle of $MPC$ is the resolution, at each sample-time, of a finite horizon optimal control problem. The optimization step helps to obtain an optimal control input sequence and the first value of this sequence is applied to the system. Then, the principle of $MPC$ is based on five steps which are performed successively at each sample time $k$:

- the prediction of the system evolution from a state $x_k$ at time $k$,
- the definition of the trajectory to be tracked in a finite horizon $T$,
- the definition of the cost criterion,
- the minimization of the cost criterion over the finite horizon $T$,
- the application of the optimal control solution first term.

**Prediction of the system evolution** The first step consists in predicting the system output using the knowledge of the system dynamics response. This knowledge helps to predict the future evolution of the system up to a limited prediction horizon $N_p$. The prediction may be done by using a linear or non-linear model, a black, gray or white box model.

**Definition of the reference** The reference trajectory corresponds to the trajectory that must be followed by the system. Two solutions help to define this reference. On the one hand, if only the final or desired point is known, a smooth reference trajectory is built based on a reference model. On the other hand, if the trajectory is a priori defined, which is the case here, the latter is directly used.

**Definition of the cost criterion** The cost criterion is commonly composed of two terms. The first corresponds to the sum of the quadratic errors between the reference trajectory and the predicted outputs of the model. The second describes the limitation of the control signal. In the case of a $SISO$ system, at time $k$, the cost criterion $J(k)$ is expressed by:

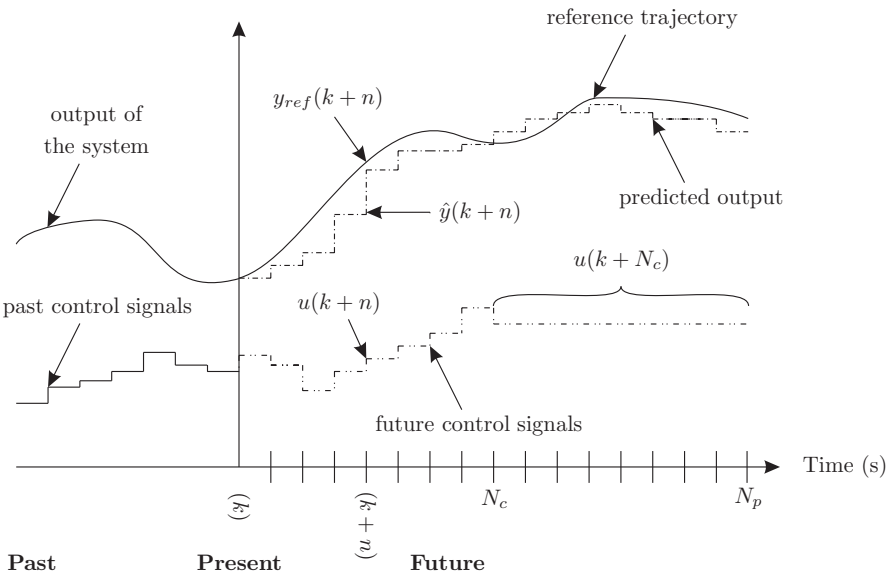



Figure 3.16: Definition of the signals used for MPC

$$J(k) = \sum_{n=1}^{N_p} |\hat{y}(k+n) - y_{ref}(k+n)|_Q + \sum_{n=0}^{N_c} |u(k+n)|_R \tag{3.40}$$

with $Q$ and $R$ the weighting factors on the output error and the control input, $\hat{y}$ the predicted output, $y_{ref}$ the reference trajectory, $u$ the control signal, $N_p$ the prediction horizon and $N_c$ the control horizon. $N_c$ is defined such that $N_c \leq N_p$ and beyond $N_c$ ($n > N_c$), $u(k+n)$ is constant and is defined by $u(k+n) = u(k+N_c)$.

**Minimization of the cost criterion** The minimization of the criterion $J(k)$ gives the optimal control input sequence. Without constraint and considering a prediction based on a linear model, it is possible to solve the optimization problem analytically by using a QR decomposition [Maciejowski, 2000]. For the general case, the minimization is described by:

$$\begin{aligned} u^* &= \min_U J(k) \\ \text{respecting} \quad C &: \begin{cases} u_{\min} \leq u(k+n) \leq u_{\max} \forall k \in [1, N_c] \\ \hat{y}_{\min} \leq \hat{y}(k+n) \leq \hat{y}_{\max} \forall k \in [1, N_p] \\ \hat{x}_{\min} \leq \hat{x}(k+n) \leq \hat{x}_{\max} \forall k \in [1, N_p] \end{cases} \end{aligned} \tag{3.41}$$

such that $\hat{x}$ corresponds to the estimated state of the system.

**Application of the first term of the optimal solution** The first term of the optimal control sequence, obtained by solving (3.40) considering (3.41), is applied on the system. Finally the different signals which are involved in the frame of $MPC$ are detailed in Fig.3.16.

#### 3.5.3.2 Conclusion

From this Lateral Controller presentation, it can be clearly seen that constraints are integrated in the Controller itself (control signals, predicted states and predicted outputs may be constrained) but also in the Reference generation (constrained trajectory generation). In addition, here again, the importance of the *Reference* is shown. Indeed, the Lateral Controller refers to a trajectory tracking problem, so tries to match the *Vehicle* position to the trajectory. If the quality of the controller is defined regarding its ability to manage the *Vehicle* aspects (*Vehicle* limitations, *Driver* comfort,etc.), it is the *Reference* which is mostly in charge of the integration of the constraints related to the *Environment* (road location, validity area, etc.). This is confirmed by the results presented in Section.5.4.

## 3.6 Conclusion

In this Chapter, the detailed description of the unconstrained and constrained trajectory generation approaches have been presented. From this description, the differences between these two approaches have been shown. The unconstrained trajectory generation, as it does not consider any of the constraints involves to deal with the constraints at the controller level. Indeed, the latter, to provide a safe and comfortable speed control, has to perform several additional tasks such as speed profile modification for instance. Contrary to this, the constrained approach, the trajectory generation as an optimization problem, directly considers limits related to the *Driver*, to the *Vehicle* and the *Environment*. If this increases the complexity level and the computation time required to their generation, trajectories are expected to be more adapted to automotive purposes.

These two trajectory generation techniques, through their description, have shown the benefits of the constraints management approach considered in this PhD. The next Chapter is consequently dedicated to the presentation of the second approach: the information combination.

# Chapter 4

# Dempster-Shafer Fusion for Speed Limit Assistant

**Contents**



## 4.1 Introduction

In Chapter.3, the description of the unconstrained and constrained trajectory generation respectively applied to the Longitudinal and Lateral Control of a car-like *Vehicle*, have been presented. This has revealed the importance of constraints management, the first approach of this PhD. This Chapter aims at describing in details the different techniques adopted for the second approach: the information combination. As mentioned earlier in this document, the considered

application is the Speed Limit Assistant ($SLA$) which aims at giving the most relevant limit speed information to the *Driver* regarding information provided by a navigation and a Speed Limit Sign Recognition ($SLSR$) systems.

In Chapter.1, the information combination has been shown to be a challenging task as sensors may provide inaccurate, redundant, erroneous, etc. information. This point is confirmed by the $SLA$ overview corresponding to the first section of this Chapter. To cope with these limitations, the $SLA$ presented in this Chapter integrates all these sensor information particularities through a two-level $DS$-based fusion: a local processing of the sensor data and a multi-sensor data fusion. This multi-level fusion, approach helps to determine the best sensor information regarding its reliability, i.e. its inaccuracies and then to provide more reliable speed limit information to the *Driver*. In addition, the navigation information is determined through a multi-criterion fusion considering that each *Digital Map Database* attribute is given by independent and specialized sources. This multi-criterion fusion helps to detect the navigation errors and selects the best speed regarding to the road context. To develop such a system, a complete refinement of the belief mass modeling and estimation has been done regarding the original $SLA$ developed in [Bradai, 2007]. This Chapter aims at describing in details these modifications further to the presentation of the Dempster-Shafer Theory basics.

## 4.2 Basics of Evidence Theory

The Evidence theory, based on the upper and lower probabilities, has been introduced by Dempster [Dempster, 1967] and mathematically formalized by Shafer [Shafer, 1976]. It may also be found to be called *Belief theory* or *Dempster-Shafer theory*. This theory is based on the modeling of the belief level one has in a defined event, through functions defined on subsets (singletons and/or unions, contrary to the Probability theory). Usually these functions, named *mass functions* (or *belief masses*), are defined in $[0, 1]$. This section only introduces the basic elements of this theory required for the considered $SLA$ application. More information are available in [Shafer, 1976].

### 4.2.1 Modeling

Let $\Theta$ be the set called *frame of discernment* of all the possible solutions (or hypotheses) $H_i, i = 1, 2, ..., k$ to a considered problem such that[19]:

$$\Theta = \{H_1, H_2, ..., H_k\} \tag{4.1}$$

with its corresponding *referential subset* $2^\Theta$ of $2^k$ propositions $A$ of $\Theta$ such that:

$$2^\Theta = \{A/A \subseteq \Theta\} = \{\emptyset, H_1, H_2, ..., H_k, \{H_1, H_2\}, ..., \{H_2, H_3\}, ..., \Theta\} \tag{4.2}$$

$\emptyset$ is the impossible hypothesis which usually characterizes the conflict between the sources, and $\Theta$ is the ignorance (i.e. the union of all the hypotheses $H_i$). The referential subset defines all the possibilities which can occur regarding to the hypotheses of the discernment frame. It can be noted that singletons but also unions are considered in the referential subset. For convenience, the notation $H_i \cup H_j$ will be preferred to $\{H_i, H_j\}$ in the following parts of the document.

The veracity of a proposition is characterized by its *basic belief assignment* ($bba$) or mass $m$ defined as follows:

[19] $k$ is the number of possible hypotheses.

$$\begin{array}{c} m : 2^{\Theta} \rightarrow [0,1] \\ \sum\limits_{A \subseteq 2^{\Theta}} m(A) = 1 \end{array} \tag{4.3}$$

and which has the following properties:

- $A \subseteq 2^{\Theta}, m(A) \neq 0 \Leftrightarrow A$ is a focal element,
- The set of focal elements is called the core $C_{\Theta}$: $C_{\Theta} = \left\{A \subseteq 2^{\Theta}, m(A) > 0\right\}$,
- $\forall A \subseteq 2^{\Theta} \neq \Theta, m(\Theta) = 1$ et $m(A) = 0 \Leftrightarrow$ total ignorance,
- $A \subseteq 2^{\Theta}, m(A) = 1 \Leftrightarrow$ the source is said to be imprecise (it only believes in proposition $A$ considering that it can be a disjunction of hypotheses),
- $H_i \subseteq \Theta, m(H_i) = 1 \Leftrightarrow$ the source is said to be precise (there is no doubt about the veracity of $H_i$),
- $A \subseteq 2^{\Theta}, m(A) = s$ and $m(\Theta) = 1 - s$, $s \in [0,1] \Leftrightarrow$ the source is said to be imprecise and inaccurate (it only has some belief in $A$)

Remark: In [Shafer, 1976], Shafer stipulates that the frame $\Theta$ has to be **exhaustive**, in other words, whatever the configuration of the data, the solution lies within the frame of discernment. In this case, the data fusion is said to be in *closed world*, so there is a mass $m(\emptyset) = 0$ on the impossible hypothesis (also called conflict). In addition, the hypotheses have to be **exclusive** - the solution is unique: $H_i \cap H_j = \emptyset, \forall i \neq j$.

However, the definition of an exhaustive and exclusive discernment frame may be a difficult task for real applications. Consequently, two other *worlds* have been suggested: the *open world* and the *extended open world*. The first one, proposed in [Smets and Kennes, 1994] defines the frame of discernment as the container of all the known hypotheses but not all the possible hypotheses. The exhaustiveness assumption is, in this case, raised. Then a mass $m(\emptyset) \geq 0$ on the impossible hypothesis is tolerated and represents a *new* or an unknown hypothesis called *reject class*. The limitation of the *open world* is that it is defined considering that the sources are all completely reliable. Furthermore, in the *open world*, the origin of the conflict is difficult to be found. Indeed, as the conflict can originate from the non exhaustiveness of the discernment frame or from the source discordance, considering the null hypothesis as a new solution can result in hazardous situations.

To cope with this limitation, [Rombaut, 1998, Gruyer, 1999, Royère, 2002] propose to use the *extended open world*. In these approaches, an additional hypothesis $*$ is added in the discernment frame in order to explicitly represent the unknown hypothesis as shown in (4.4). This singleton allows the frame of discernment to become exhaustive ($*$ is exclusive w.r.t. the other hypotheses) with non-totally reliable sources. In addition, as the non-exhaustiveness is solved by the addition of an hypothesis, a non zero value of the null hypothesis mass only refers to the conflict originated by the sources discordance. However, this hypothesis cannot be related to any sources as it is not modeled and requires the necessity to redefine the hypothesis complementarity ($H_i^c$) such as presented in (4.5).

$$\begin{array}{c} \Theta = \{H_1, H_2, ..., H_k, *\} \\ \text{with: } 2^{\Theta} \{\emptyset, H_1, H_2, ..., H_k, *, H_1 \cup H_2, ..., H_i \cup *, ..., \Theta\} \end{array} \tag{4.4}$$

$$H_i^c = \{H_1 \cup H_2 \cup ... \cup H_{i-1} \cup H_{i+1} \cup ... \cup H_k \cup *\} \tag{4.5}$$

### 4.2.2 Estimation

As mentioned in Chapter.1, the *Estimation* step depends on the selected formalism. The most common approach is to affect the mass $m_j$ of a source $S_j$ on a non-empty proposition $A$ of $2^\Theta$ [Shafer, 1976] such that:

$$\begin{cases} m_j(A) = s \\ m_j(\Theta) = 1 - s \\ m_j(B) = 0, \forall B \in 2^\Theta, B \neq A, B \neq \Theta \end{cases} \tag{4.6}$$

with the real $s \in [0, 1]$. Note that this method affects the mass of the source to only one proposition (which could be a singleton or an union of hypotheses) and to the ignorance. Thus, if $s = 0$, then $m_j(\Theta) = 1$ so defining a total ignorance for source $S_j$. A similar approach, presented by [Yager, 1987] affects the mass $m_j$ of $S_j$ on a non-empty proposition $A$ of $2^\Theta$ and on its complementary proposition $A^c$ as follows:

$$\begin{cases} m_j(A) = s \\ m_j(A^c) = 1 - s \\ m_j(B) = 0, \forall B \in 2^\Theta, B \neq A, B \neq A^c \end{cases} \tag{4.7}$$

A more interesting method has been initially proposed in [Rombaut, 1998] and described by [Gruyer, 1999] and [Royère, 2002]. This method is based on the consideration of specialized sources: a source can only give information about one specific proposition of the discernment frame. Consequently, the specialized source $S$ can only say that *"It is the proposition"* $(H_i)$, *"It is not the proposition"* $(H_i^c)$ and *"I do not know"* $(\Theta)$. Consequently the method is based on the consideration of the triplet $\{H_i, H_i^c, \Theta\}$, respectively the hypothesis, the complementary hypothesis and the ignorance. Then, the axiom that *"a source cannot simultaneously affect mass on the hypothesis and its complementary, thus giving antagonistic information"* implies the masses to be described by (4.8) and illustrated in Fig.4.1.

$$\begin{aligned} m_j(H_i) &= \begin{cases} 0 & C_v \in [0, \tau] \\ \Phi_1(\alpha_0, C_v) & C_v \in [\tau, 1] \end{cases} \\ m_j(H_i^c) &= \begin{cases} \Phi_2(\alpha_0, C_v) & C_v \in [0, \tau] \\ 0 & C_v \in [\tau, 1] \end{cases} \\ m_j(\Theta) &= \begin{cases} 1 - \Phi_2(\alpha_0, C_v) & C_v \in [0, \tau] \\ 1 - \Phi_1(\alpha_0, C_v) & C_v \in [\tau, 1] \end{cases} \end{aligned} \tag{4.8}$$

with the following limit conditions:

$$\begin{cases} C_v = 0 & \Phi_2(\alpha_0, 0) = \alpha_0 \\ C_v = \tau & \Phi_2(\alpha_0, \tau) = 0 \\ C_v = \tau & \Phi_1(\alpha_0, \tau) = 0 \\ C_v = 1 & \Phi_1(\alpha_0, 1) = \alpha_0 \end{cases} \tag{4.9}$$

with a real $\tau \in [0, 1]$, $\Phi_1$ and $\Phi_2$ two functions describing the evolution of the masses regarding to $\alpha_0$ the maximum mass value, and $C_v$ a confidence variable. The latter helps to define the confidence level which is referred to the source $S_j$ and may be of different composition: information coming from another sensor, from an external source, from an expert, a combination of these examples, etc.

As it can be seen in Fig.4.1, the present approach is characterized by mass functions avoiding the overlapping of antagonistic hypotheses. If $\tau < 0.5$, then the estimation is said to be optimistic

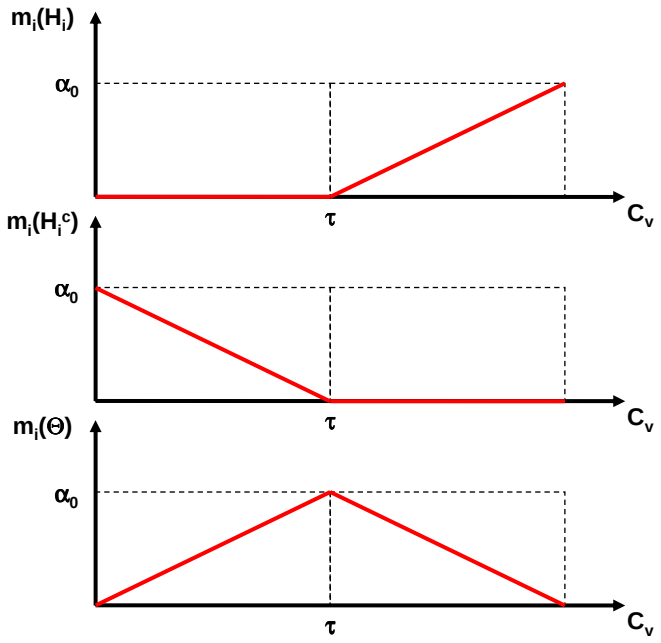



Figure 4.1: Mass Estimation in [Rombaut, 1998]

(the mass $m_j(H_i)$ of a given hypothesis $H_i$ is $\geq 0$ if $\tau < 0.5$), and in the opposite, if $\tau > 0.5$ the estimation is said to be pessimistic (if $\tau = 0.5$ the estimation is said to be neutral). Finally, the non overlapping property of this estimation avoids the generation of conflict.

Other solutions for the *Estimation* step are: singleton functions when the fusion process is only defined by singletons [Lefevre et al., 1999], probabilistic models [Appriou, 2001], distance models [Denoeux, 1995], or other approaches such as presented in [Rombaut and Zhu, 2002].

### 4.2.3 Combination

The *Combination* step is the core of the fusion as it gathers and regroups the different information of the sources $S_i$ using a combination operator[20]. The choice of the combination operator, as it has a direct impact on the fusion results, has to be done wisely. The most common combination operator is the Dempster/Shafer ($DS$) conjunctive operator $\bigoplus$:

$$m_{1\ldots k}(H) = m_{S_1}(H) \oplus m_{S_2}(H) \oplus \ldots \oplus m_{S_k}(H) = \sum_{H_1 \cap \ldots \cap H_k = H} \prod_{i=1}^{k} m_{S_i}(H_i) \tag{4.10}$$

with $k$ the number of sources and $m_{1\ldots k}(H)$ the unique mass estimation over the hypothesis $H$ resulting from the combination. This DS operator can only be used when sources are independent as it is not idempotent ($m \bigoplus m \neq m$). For example, the application of (4.10) for an hypothesis $H$ described by two sources $m_{S_1}$ and $m_{S_2}$ is defined by:

$$m(H) = \sum_{A_i \cap B_j = H} m_{S_1}(A_i) \cdot m_{S_2}(B_j) \tag{4.11}$$

[20] Note that each source has its own mass estimation $m_{S_i}$ defined on the common frame of discernment $\Theta$.

Table 4.1: Masses of Zadeh's Example

| | $H_1$ | $H_2$ | $H_3$ |
|---|---|---|---|
| $m_{S_1}$ | 0.9 | 0.0 | 0.1 |
| $m_{S_2}$ | 0.0 | 0.9 | 0.1 |

The generalization of (4.10) to $k$ sources can be considered as an iteration of the combination of two sources (4.11) such as:

$$m = m_{S_1} \oplus m_{S_2} \oplus ... \oplus m_{S_k} = ((((m_{S_1} \oplus m_{S_2}) \oplus m_{S_3}) \oplus ...) \oplus m_{S_k}) \tag{4.12}$$

This is possible as the *Dempster* operator is associative and commutative, so does not force the definition of a fusion order over the $k$ sources. Nevertheless, a larger number of sources combined with a large frame of discernment may result in a time consuming fusion process. In addition, this operator may generate conflict ($m(\emptyset) \neq 0$) which is in contradiction with the *closed world* consideration. This introduces the problem of *Conflict Management*.

### 4.2.4 Conflict Management

Conflict is an important element of the *Data Fusion* process which can describe the non-exhaustiveness of the discernment frame, the sources discordance or the unreliability of these sources. Its particularity is to be absorbing; in other words, it tends to increase with the number of combinations. The management of the conflict is consequently of a great importance and can be done using four main types of solutions: redistribution, discounting, use of particular combination operators, or use of conflict as a source of information. The list of solutions presented here is not exhaustive and focuses on the main approaches developed in the literature.

- Conflict redistribution: a total redistribution of the conflict represented by $m(\emptyset)$ on all the other propositions of the discernment frame is done and implies a null mass on the impossible hypothesis ($m(\emptyset) = 0$). This fulfills the constraints involved by the *closed world*. The most common redistribution technique, known as the *Normalization* [Shafer, 1976], is based on the multiplication of the masses $m(H)$ obtained after combination, by a factor $\kappa$ such that:

$$\begin{aligned} m(H) &= \kappa \cdot \sum_{A_i \cap B_j = H} m_{S_1}(A_i) \cdot m_{S_2}(B_j) \\ with: \kappa &= \frac{1}{1 - m(\emptyset)} = \frac{1}{1 - \sum_{A_i \cap B_j = \emptyset} m_{S_1}(A_i) \cdot m_{S_2}(B_j)} \end{aligned} \tag{4.13}$$

  The limitation of this solution is that it redistributes the conflict equally on all the masses whatever the conflict level increasing the belief mass of the hypotheses given by both sources. This limitation has been clearly shown by [Zadeh, 1979]. Indeed, let us consider two sources $S_1$ and $S_2$ which have masses on three singleton hypotheses $H_1$, $H_2$ and $H_3$ as presented in Table.4.1. It can be easily seen that $S_1$ and $S_2$ are in conflict since $S_1$ believes in $H_1$ and $S_2$ in $H_2$. The combination results, presented in Table.4.2 clearly shows that the normalization involves an aberrant increase of the mass on $H_3$, so that the belief in $H_3$ goes from 0.01 for the non-normalized case to 1.0 with normalization.

Table 4.2: Normalization impact

| | $H_1$ | $H_2$ | $H_3$ | $\emptyset$ |
|---|---|---|---|---|
| $m_{S_1} \oplus m_{S_2}$ not normalized | 0.0 | 0.0 | 0.01 | 0.99 |
| $m_{S_1} \oplus m_{S_2}$ normalized | 0.0 | 0.0 | 1 | 0.0 |

To reduce these effects, other redistribution principles have been developed such as in [Lefevre et al., 2000] or in [Florea et al., 2006]. These two solutions are based on the proportional redistribution of the conflict on the propositions generating it. With these techniques, the redistribution seems to be more appropriate, however, they do not solve the problem of overconfidence in cases of high conflict redistribution.

Another approach presented by *Yager* [Yager, 1987] tends to affect the conflict on the ignorance ($m(\Theta)$). This solution allows to fulfill the constraints of the *closed world* and to avoid the overconfidence problem in case of high conflict. Indeed, in this particular case, the fusion, which was originally characterized by a high conflict, is characterized by a high ignorance after the redistribution. It does not involve any changes on the proposition masses, thus conserving their initial information.

Note that, these solutions are mostly applied to eliminate the conflict, thus to meet the requirement of *closed world*, but can also be adapted to the other worlds.

- Discounting: as mentioned in the previous section, during the *Modeling*, additional information can be used. A relevant information could be the quantification of the source reliability which may define the level of confidence given to a source. This can be modeled in the Evidence Theory through *Discounting* factors [Shafer, 1976] based on the weighting of the source's mass such that:

$$\begin{cases} m'(A) = \alpha.m(A) \\ m'(\Theta) = 1 - \alpha(1 - m(\Theta)) \end{cases} \tag{4.14}$$

with $\alpha \in [0, 1]$ the weighting coefficient of the considered source. Note that $\alpha = 0$ and $\alpha = 1$ respectively denotes a completely unreliable and completely reliable source. This solution reduces the conflict induced by information provided by unreliable sources through the reduction of their respective mass on the considered hypothesis.

- Combination operators: the use of different combination operators can also be a solution. Let us consider the disjunctive combination operator presented in [Smets, 1993]. Contrary to the $DS$ operator, this one is no longer considering hypotheses intersections but hypotheses unions:

$$m(H) = \sum_{H_1 \cup \ldots \cup H_k = H} \prod_{i=1}^{k} m_i(H_i) \tag{4.15}$$

As the mass distribution is performed on unions of propositions, which usually includes the discordant source information, conflict cannot be generated. However, this solution, as it dilutes the information into unions, is not as precise as the conjunctive operator. To keep the benefits of each operator, [Dubois and Prade, 1988] proposed an hybrid operator named conjunctive/disjunctive operator. The latter is described by the succession of a conjunctive

and a disjunctive step: a conjunctive step is performed to determine precisely the masses on the hypotheses and then, if conflict is generated, a disjunctive step is processed to redistribute it on unions of propositions. This solution is consequently a good compromise between precision and reliability and is defined by (4.16).

$$m(H) = \sum_{H_1 \cap \ldots \cap H_k = H} \prod_{i=1}^{k} m_i(H_i) + \sum_{\substack{H_1 \cup \ldots \cup H_k = H \\ H_1 \cap \ldots \cap H_k = \emptyset}} \prod_{i=1}^{k} m_i(H_i) \tag{4.16}$$

- Conflict as an additional information: indeed, by considering that the frame of discernment is exhaustive, the conflict describes the discordance between sources. The origin of this discordance can be of different types (a source false detection, a source malfunctioning, a different interpretation, etc.), thus can be interpreted as an information for the data fusion [Rominger and Martin, 2010].

### 4.2.5 Decision

The *Decision* is the last step in the *Data Fusion* process. It consists in the selection of the most relevant solution - a single hypothesis ($H_i$) or a union of hypotheses ($H_i \cup H_j$) - regarding to the frame of discernment and to a criterion. This step, as it selects one solution in a set, involves a loss of information (the information stored in the non selected solutions). There are three main decision criteria: maximum of *belief* or *credibility*, maximum of *plausibility* and maximum of *bet probability*.

- *Decision* can be taken by considering the hypothesis $H_i$ which has the maximum value of *belief function* such as:

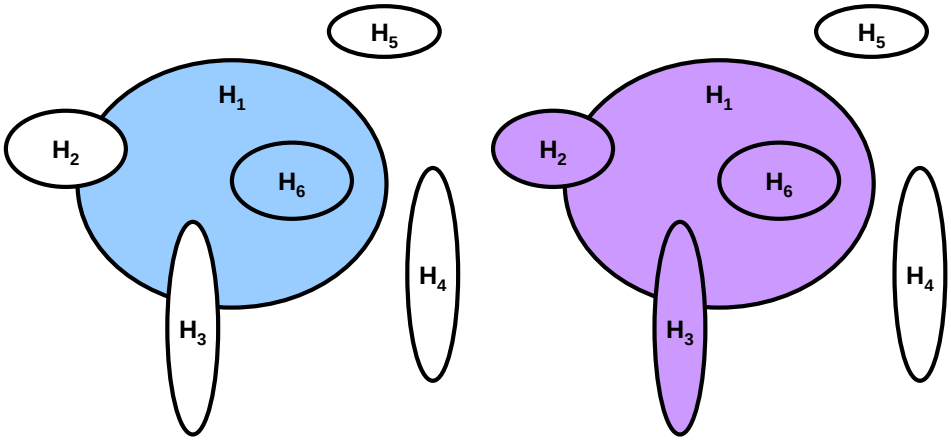
$$\begin{aligned} &Bel\,(H_i) = \max_{1 \leq j \leq k} Bel\,(H_j) \\ &with : Bel\,(H_j) = \sum_{A \subseteq H_j} m\,(A) \end{aligned} \tag{4.17}$$

  This solution is said to be pessimistic as only the propositions included in $H_j$ are considered for the determination of the belief function. In other terms, it corresponds to the mass sum of all the hypotheses which are exclusively contained in $H_j$ as presented in Fig.4.2.a.

- Contrary to the maximum of *belief*, the maximum of *plausibility* is said to be optimistic. Indeed, it corresponds to the mass sum of all the hypotheses having a non-null intersection with $H_j$ as presented in Fig.4.2.b:

$$\begin{aligned} &Pl\,(H_i) = \max_{1 \leq j \leq k} Pl\,(H_j) \\ &with : Pl\,(H_j) = \sum_{A \cap H_j \neq \emptyset} m\,(A) = 1 - Bel\left(H_j^c\right) \end{aligned} \tag{4.18}$$

- The maximum of *pignisitc probability* introduced by [Smets, 1990], is a compromise between these two criteria as it distributes proportionally the masses of proposition unions on singletons which compose it. Although, the use of this prudent criterion requires a combination operator which possesses the associativity property:




(a) Belief Representation (b) Plausibility Representation

Figure 4.2: Belief vs Plausibility of $H_1$

$$BetP_{\Theta}(H_i) = \sum_{\substack{A \in 2^{\Theta} \\ H_i \subseteq A}} \frac{1}{|A|} m(A) \tag{4.19}$$

with $|A|$ the cardinality of $A$. In this case, the solution $H_i$ can only be a singleton whereas in the two previous cases, $H_i$ can also be an union of propositions.

## 4.3 Speed Limit Assistant Overview

*SLA* systems help in the determination of the current legal speed limit. Their goal is then to help the *Driver* in its perception task. The detection of speed signs represents a challenging task as speed limit signs are usually specific for each country, so may be of different value, different color, different shape, etc. Speed Limit Assistants have then to be based on robust and reliable sensors.

The first generation of Speed Limit Assistant aimed at the detection of the traffic signs during the driving task with a camera coupled to a *SLSR* algorithm. *SLSR* emerged in the 1980's and have become a widely studied research field. They are currently mainly decomposed in three sub-processes [Gavrila, 1999]: the detection, the recognition and the time integration - more information about *SLSR* can be found in [Bargeton, 2009]. However, even if great advances have been done in *SLSR* algorithms, involving improvements of the speed limit information, they were based on a lone camera which is subject to limitations. The main problem is the impact of the weather context or the impact of night which usually degrades the image quality provided by cameras, and consequently decrease the quality of speed limit sign recognition.

To overcome these limitations, a new generation of *SLAs* have emerged: multiple-sensor-based *SLAs* which usually refers to the combination of the information provided by a *SLSR* with *Digital Map Database* information. Indeed, *SLSR* and navigation systems are complementary as presented in Table.4.3. Nevertheless their information combination can be processed

Table 4.3: Advantages and Limitations of Camera and *Digital Map Database*

| *Sensor* | *Advantage* | *Limitations* |
|---|---|---|
| Camera | Real-time traffic sign detection (up-to-date), temporary traffic sign detection, country specific traffic detection | efficiency reduction in bad environmental conditions, in night driving, may not detect all the traffic signs (shape, color, etc.), still subject to false detections |
| Digital Map Database | Provide speed limit independently of the sign shape or color, provide speed limit independently of the environmental conditions, operational during nights | May not be completely up-to-date, may not contain country road traffic signs, may contain erroneous speed limits |

using different techniques. Here the focus is put on combinations based on Data Fusion techniques and more precisely on techniques based on the Evidence Theory. This formalism has been used in [Nienhüser et al., 2009] which proposes to use contextual information in order to characterize the quality of the vision and navigation information. This helps to reduce the weight of navigation information in case of road maintenance or to reduce the weight of vision information during night-driving for instance. However, sensor weight levels are obtained by performing additional camera image processing. The independence of the sources may not be guaranteed as the vision information acts the navigation information. [Lauffenburger et al., 2008] proposed a different approach in which the navigation information is processed through the combination of several criteria corresponding to *Digital Map Database* attributes. These attributes are of great help in the description of the current road context, thus allows to characterize the confidence which can be put in the speed returned by the digital map. However, this approach does not allow to consider all the navigation system inaccuracies and do not use efficiently the criteria. Finally, its multi-sensor combination (between the $SLSR$ and the navigation information) was done by giving more importance to the vision information. To overcome these limitations, [Puthon et al., 2010] proposed an enhancement of [Lauffenburger et al., 2008] navigation information estimation. This improvement is obtained through a reconsideration of the criteria regarding their nature. However, this approach is still subject to limitations.

Contrary to these approaches, the present $SLA$ does not consider the navigation information through weighted sums but through a multi-criterion fusion which takes the quality of the navigation system into account, i.e. the quality of the positioning, the localization and the *Digital Map Database*. This present the great advantage to detect the errors and to determine the best navigation speed more efficiently. Furthermore, the multi-sensor fusion has also been upgraded so that each sensors has yet the same importance.

## 4.4 Multi-Level Fusion Approach: a Multi-criterion Multi-sensor Fusion

The strategy presented in Fig.4.3 is based on the multi level approach described in Chapter.1. This figure clearly shows a two steps fusion: one for the determination of the sensors information

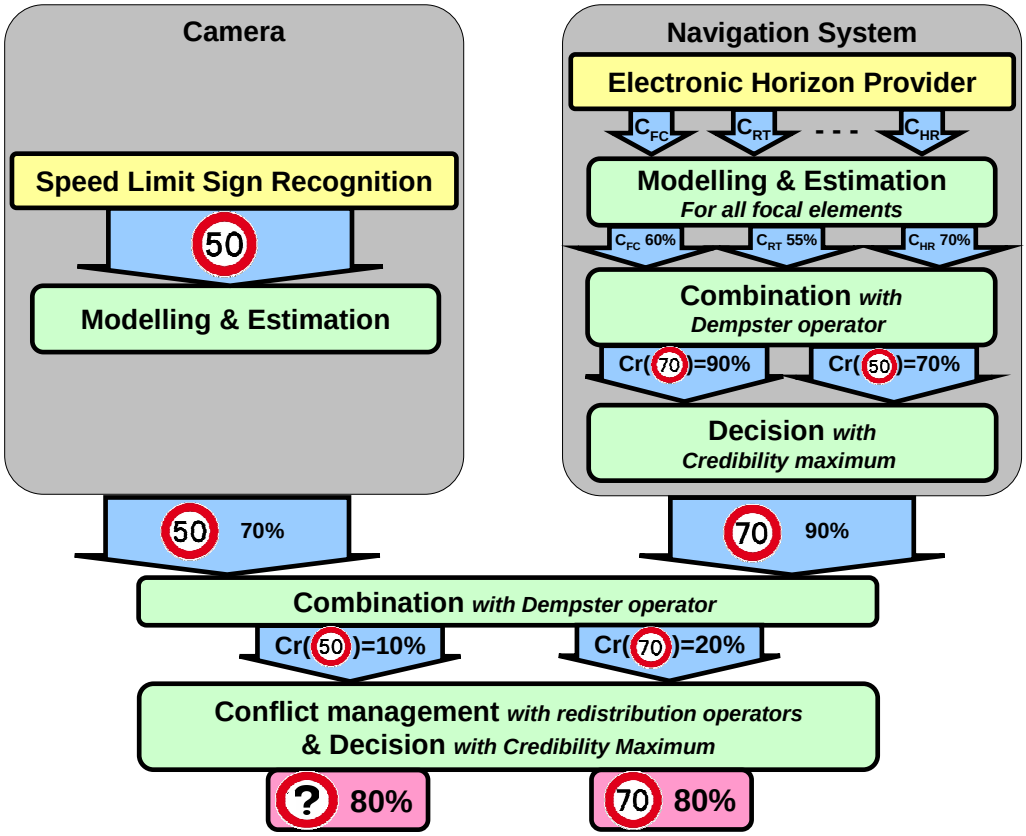



Figure 4.3: Multi-criterion and Multi-sensor Fusion for Speed Limit Determination

and the second dedicated to the multi-sensor fusion[21].

The first step of the global fusion consists in the processing of the sensor data (information coming out of the gray blocks), i.e. the determination of each sensor speed and its related belief. For the vision, it consists in the belief mass estimation of the information provided by the Speed Limit Sign Recognition algorithm. On the opposite, the navigation presents a more complex structure: it consists in a first fusion step based on digital map attributes named here criteria. This multi-criterion fusion helps to determine the best navigation speed over a predefined set, its belief mass, and may detect false navigation information. Indeed, the multi-criterion approach helps to characterize more precisely the different solutions of the given problem as introduced by [Royère, 2002, Mourllion, 2006] and presented in Fig.4.4. This figure presents two speeds $S_1$ and $S_2$ characterized by three criteria $C_1$, $C_2$ and $C_3$ which are considered to be independent. It is clear that, considering $C_1$ and $C_2$, $S_1$ and $S_2$ are similar. The consideration of $C_3$ is consequently essential to the discrimination of both speeds. This multi-criterion approach marks its difference with the aforementioned research studies which does not consider the multi-dimensionality of the navigation information, i.e. the consideration of each criterion as an independent and a specialized source of information.

If the proposed multi-criterion approach helps to provide more accurate navigation information, it also helps in the detection of incoherent navigation information. Indeed, consider the following example: navigation speed is $50km.h^{-1}$, $C_1 > 0.6$, $C_2 = ADASAttributeActivated$, $C_3 = FunctionalClassIs0$, $C_4 = Highway$, $C_5 = InCityDriving$, $C_6 = NoIntersectionDetected$,

[21] Corresponding to the fusion of the vision and the navigation information.

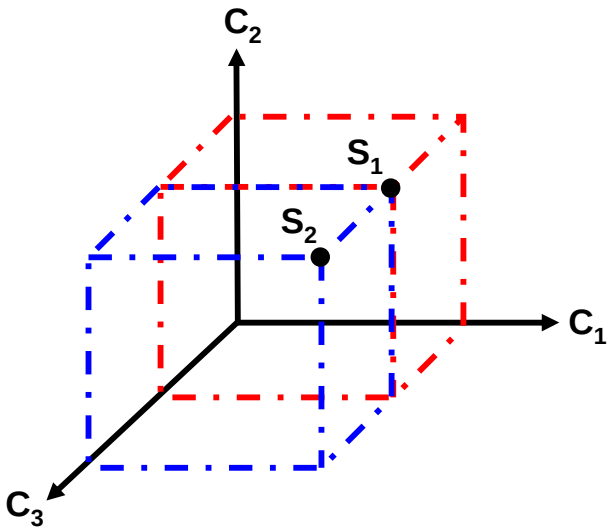



Figure 4.4: Information in the Criteria Point of View [Mourllion, 2006]

$C_7 = HighwayRampDetected$, and $C_8 = GuidanceModeActivated$. Considering the weighted sum approach presented in [Lauffenburger et al., 2008], the navigation belief mass for the speed of $50km.h^{-1}$ ($m_{n=50}(50)$) is calculated as follows:

$$m_n(50) = \frac{\alpha_1 \cdot 0.9 + \alpha_2 \cdot 0.9 + \alpha_3 \cdot 0.9 + \alpha_4 \cdot 0.2 + \alpha_5 \cdot 0.9 + \alpha_6 \cdot 0.6 + \alpha_7 \cdot 0.8 + \alpha_8 \cdot 0.9}{\sum_{i=1}^{8} \alpha_i} = 0.71 \tag{4.20}$$

This example shows the limitation of the weighted sum approach. Indeed, the criteria are contradictory as the *Vehicle* drives in a city while being on a highway which functional class is validated (for high importance road, the functional class is validated) and with the detection of a highway ramp. There should be an error in the database as these criteria are incoherent. However the belief in the navigation is still high as its mass is equal to 0.71. A solution to this problem can be to redefine the strategy adopted for the determination of the weighting factors or to separate the criteria for a multiple weighted sum calculation as presented in [Puthon et al., 2010]. However, this solution, may still not be able to detect correctly the *Digital Map Database*, the positioning and the localization errors. To cope with this problem, the current approach considers a multi-criterion fusion. If the multi-criterion approach helps to detect and correct the navigation speed errors, it also considers the positioning and localization inaccuracies, so provides an information which considers all the navigation system errors.

After the navigation multi-criterion fusion, a multi-sensor combination is performed. In the previous Speed Limit Assistant, the multi-sensor fusion (navigation + speed sign recognition) was processed on all the focal elements provided by the navigation. If this helped to provide a solution for almost all the encountered situations, it was giving a higher weight to the vision information. In fact, if the indicated navigation speed was different from the vision speed, the system was looking for a focal element of the navigation which was corresponding to the vision speed. If one could be found, the result of the fusion was automatically this common speed. As the new strategy detects and corrects the speed of the navigation through the multi-criterion approach, the navigation is now only giving one speed instead of all focal speeds. This presents

the advantage to give the same importance to both sensors.

The proposed strategy also provides a conflict management step which includes the *Dempster* normalization, in addition to new approaches described in the next sections. Indeed, the conflict is now considered in two simultaneous processes (cf. Fig.D.1):

- It serves as additional information in the *Decision* step of the multi-sensor fusion. Indeed, its presence usually refers to a sensor false information or to the non-exhaustiveness of the discernment frame. Its value has consequently to be considered when the selection of the final speed limit is done (multi-sensor *Decision* step).
- It will be shown in Section.4.7 that the consideration of the conflict as an additional information increases the number of situations in which the multi-sensor fusion is not able to determine the final speed limit (so gives an *undefined speed*). Indeed, if the conflict is too high, no decision can be taken. To provide an indication about the most probable speed limit, a conflict redistribution approach is then considered in parallel to the principal fusion process. Different redistribution operators have been evaluated (including the *Dempster* normalization). However, the retained redistribution operation is the *Florea* operator [Florea et al., 2006].

The next sections are dedicated to the detailed description of the different enhancements.

## 4.5 Reliant Belief Masses Modeling and Estimation

The modeling and estimation steps are the most critical steps in data fusion. Indeed, if the model is not consistent (non exhaustive, non exclusive, etc.), the estimation of the masses is biased, and the combination will not give coherent results and will generate conflict whatever the adopted strategy. These steps have consequently to be designed wisely.

### 4.5.1 Navigation Criteria Selection

In the fusion process presented in [Lauffenburger et al., 2008], the criteria are extracted from the navigation system (through the *Electronic Horizon Provider*) and their numerical values are directly determined using a pre-defined table presented. Then the weighted sum (2.1) is used to determine the mass which is allocated to the navigation speed. The example given in Section.4.4 has shown the limitations of the previous approach. A solution to this problem can be to redefine the strategy adopted for the determination of the weighting factors or to separate the criteria for a multiple mass calculation as presented in [Puthon et al., 2010]. However, this solution, which is also based on a weighted sum, may still not be able to detect correctly the *Digital Map Database*, the positioning and the localization errors. To cope with this problem, a multi-criterion fusion is proposed here. Indeed, regarding the criteria presented in these studies, several remarks can be done:

- $C_1$ and $C_2$ are respectively describing the confidence in the localization (fusion of the GPS/odometric/inertial information with digital map data) and in the digital map data. In other words, they describe the quality of two over three of the major navigation system elements which are the positioning, the localization and the digital map data. They are not directly describing elements of the road context and thus are not directly informing about the speed limit of a given driving area. Theses criteria would be more suited to the definition of the reliability of the navigation as proposed by [Puthon et al., 2010]. This

Table 4.4: Criteria Role Definition

| Data Name | Nature | Initial Role | Role |
|---|---|---|---|
| $MLCP$ value | Localization quality indicator | $C_1$ | Localization quality indicator $\Rightarrow C_{v_{nav}}$ |
| $ADAS$ Attribute | Digital map quality indicator | $C_2$ | Digital map quality indicator $\Rightarrow C_{v_{nav}}$ |
| Functional Class | Road importance indicator | $C_3$ | $C_{FC}$ |
| Road Type | Road type indicator | $C_4$ | $C_{RT}$ |
| City Driving | Road context indicator | $C_5$ | $C_C$ |
| Intersection | Road context indicator | $C_6$ | $C_I$ |
| Highway Ramp | Road context indicator | $C_7$ | $C_{HR}$ |
| Guidance Mode | Driving mode indicator | $C_8$ | Removed |

reliability level is obtained by calculating the navigation confidence variable $C_{v_{nav}}$, which helps to determine the navigation masses (cf. Section.4.5.2 and Section.4.5.4).

- $C_3$ is related to the importance of the roads. Its validation usually defines national and highway roads, so discriminates roads which may have different speed limits. This criteria is then interesting to define the driving context and its corresponding speed limit. Nevertheless, due to a *Digital Map Database* change, $C_3$ has been re-adapted (now defined regarding 5 different levels against three in the previous database) and renamed as $C_{FC}$.

- $C_8$ has been completely removed as it does refer neither to any information related to the road context nor to the qualification of the navigation reliability.

- Criteria $C_5$, $C_6$ and $C_7$ are linked to situation and contextual information helping directly in the definition of the relevant speed limit. They correspond to three elements which are very important for the determination of the speed limit: the presence of a city ($C_5$), of an intersection ($C_6$) and of a highway ramp ($C_7$). For example, if the *Vehicle* is driving in-city, the most probable speed will be $50km.h^{-1}$ [22]. They have consequently been kept and renamed respectively as $C_C$, $C_I$ and $C_{HR}$.

- Note that $C_4$ has been renamed in $C_{RT}$.

The present strategy which has redefined the role of each navigation data is more coherent regarding to their respective meaning as presented by Table.4.4. From this table, it can be seen that $C_1$ and $C_2$ are no more used for the multi-criterion fusion to best match their meaning: since they are informing about the reliability of the navigation system, they will be associated to compute the reliability factor of this sensor. Then, note that each data which is related to the road context is still considered as a criterion as it helps to conclude about the best speed limit. Finally, the guidance mode has been removed as it does not characterize the quality of a navigation component and as it does not refer to any road context information.

[22] Usual in-city speed limit on French roads.

Table 4.5: Combination of Two Specialized Sources

| | $m_{S_2}(H)$ | $m_{S_2}(H^c)$ | $m_{S_2}(\Theta)$ |
|---|---|---|---|
| $m_{S_1}(H)$ | $H$ | $\emptyset$ | $H$ |
| $m_{S_1}(H^c)$ | $\emptyset$ | $H^c$ | $H^c$ |
| $m_{S_1}(\Theta)$ | $H$ | $H^c$ | $\Theta$ |

### 4.5.2 Knowledge Modeling and Specialized Sources

The model initiated by *Rombaut* and extended by *Gruyer* introduced in Section.4.2 has been retained for the representation of the masses for the navigation information and for the vision information as it best suits the considered context. This model is based on the consideration of specialized sources: a source can only give information about one specific proposition of the discernment frame. Consequently, the specialized source can only say that *"It is the proposition"*, *"It is not the proposition"* and *"I do not know"* which respectively corresponds to $H_i$, $H_i^c$ and $\Theta$. In the present context, this means that each source can only say *"It is this speed"*, *"It is not this speed"* or *"I do not know"*. For the vision information (provided by the camera and the $SLSR$), this reasoning is straightforward as this sensor gives only one speed, the most probable one, with a reliability value about the sign recognition. The latter may therefore be used as the confidence variable of the vision (named $C_{v_{vis}}$ in the following parts of this PhD) used to determine the vision mass.

For the navigation, the reasoning is a little bit different as the speed extracted from the *Digital Map Database* involves a set of possible speeds (focal speeds), i.e. the possible speeds according to the navigation speed, which are characterized by the different criteria described in the previous section. It has been shown in the example of Section.4.4, that criteria can be incoherent with the navigation speed, thus has revealed that criteria sources are independent. Then, as each criterion source gives information about only one speed at a time (cf. Fig.4.8), they are specialized on this speed, thus implying the criteria sources to be independent **and** specialized. The *Rombaut/Gruyer* model is consequently well adapted for the current belief mass modeling.

Another point of the *Rombaut/Gruyer* model is that a source cannot say at the same time: *It is the proposition* and *It is not the proposition*. This non-overlapping of antagonistic information allows the combination to be done without generating any conflict. Indeed, let consider the combination of two specialized sources $S_1$ and $S_2$ using this model. The definition of the conjunctive combination, on discernment frame $\Theta = \{H, H^c\}$[23] is described in Table.4.5 and results in the following equations:

$$\begin{array}{ll} m(H) & = m_{S_1}(H) \cdot (m_{S_2}(\Theta) + m_{S_2}(H)) + m_{S_1}(\Theta) \cdot m_{S_2}(H) \\ m(H^c) & = m_{S_1}(H^c) \cdot (m_{S_2}(\Theta) + m_{S_2}(H^c)) + m_{S_1}(\Theta) \cdot m_{S_2}(H^c) \\ m(\Theta) & = m_{S_1}(\Theta) \cdot m_{S_2}(\Theta) \\ m(\emptyset) & = m_{S_1}(H) \cdot m_{S_2}(H^c) + m_{S_1}(H^c) \cdot m_{S_2}(H) \end{array} \tag{4.21}$$

Considering these equations, the conflict can only be generated when $m_{S_1}(H)$ and $m_{S_2}(H^c)$ are positive or when $m_{S_1}(H^c)$ and $m_{S_2}(H)$ are positive. This can never happen with the specialized source model as shown by Fig.4.1. Indeed, as the belief masses for the criteria are defined regarding the same confidence variable $C_{v_{nav}}$, they can only be defined on $H$ **or** on $H^c$.

[23] the corresponding referential subset is then: $2^\Theta = \{\emptyset, H, H^c, \Theta\}$

Finally, it is important to note that the considered model is defined by three parameters which are:

- The confidence variable $C_v$: used to define the mass value. This indicator defines the confidence which can be given to a sensor information i.e. its reliability. Its calculation is also of great importance as it refers to the inaccuracies of the sensor. For the navigation, this corresponds to the positioning, localization and digital map data inaccuracies presented in Section.2.3.2 and in Section.2.3.3.
- The definition of the belief masses function describing their evolution w.r.t. $C_v$. This can be done using numerous solutions such as quadratic polynomials, Gaussian curves, etc. In the current approach, linear expressions have been retained as presented in Fig.4.5.
- The variable boundary $\tau$ which defines the character of the belief in the sources (optimistic, pessimistic, neutral).

### 4.5.3 Basic Belief Assignment

In the considered model presented in Fig.4.5, two areas can be determined regarding the confidence variable $C_v$: the first one defined by $C_v < \tau$, in which the confidence variable is low. This area is here described by a linear decrease of the mass over the complementary-hypothesis $H_j^c$ and an increasing ignorance (there is no mass on the hypothesis $H_j$). In the second area, defined for larger confidence variable values, the mass over the hypothesis $H_j$ linearly increases and the mass over the ignorance decreases (there is no mass on the complementary-hypothesis $H_j^c$). In the literature, the extreme values for these linear functions are commonly set to 0 and 1 describing a total unbelief or a total belief in a hypothesis. Considering that our sources are never completely reliable or never completely unreliable, these extreme values can not be considered here. Consequently, in the present multi-criterion fusion, the extreme values of these functions are related to the identified criteria values, which provide a more dynamic and more flexible basic belief assignment regarding the quality of the sensor information. This solution is described by Fig.4.5 with $Crit_v$ being the considered criterion value.

For example, consider a navigation speed of $50km.h^{-1}$ and a criterion $C_{RT} = Communal$. Chapter.5 will show that this corresponds to a criterion value of $Crit_v = C_{RT,50} = 0.9$. Then using the general formulation of *Rombaut* given in (4.8), the belief mass representation for this criterion with the navigation confidence value $C_{v_{nav}}$, becomes:

$$\begin{aligned}
m_{RT,50}(50) &= \begin{cases} 0 & C_{v_{nav}} \in [0,\tau] \\ \left(\frac{0.9}{1-\tau}\right) C_{v_{nav}} - \frac{0.9\tau}{1-\tau} & C_{v_{nav}} \in [\tau,1] \end{cases} \\
m_{RT,50}(50^c) &= \begin{cases} -\frac{0.9}{\tau} C_{v_{nav}} + 0.9 & C_{v_{nav}} \in [0,\tau] \\ 0 & C_{v_{nav}} \in [\tau,1] \end{cases} \\
m_{RT,50}(\Theta) &= \begin{cases} \frac{0.9}{\tau} C_{v_{nav}} + 0.1 & C_{v_{nav}} \in [0,\tau] \\ -\left(\frac{0.9}{1-\tau}\right) C_{v_{nav}} + \frac{1-0.1\tau}{1-\tau} & C_{v_{nav}} \in [\tau,1] \end{cases}
\end{aligned} \tag{4.22}$$

It is important to note that, as the sum of the different propositions mass ($m_{RT,50}(50) + m_{RT,50}(50^c) + m_{RT,50}(\Theta)$) has to be, by definition, normalized to 1, the ignorance mass is adjusted regarding to $Crit_v$. In addition, both sensors are here defined using a neutral model. This means that $\tau = 0.5$ for the vision and the navigation basic belief assignment even if the latter is defined by a conservative confidence variable as presented in the next section.

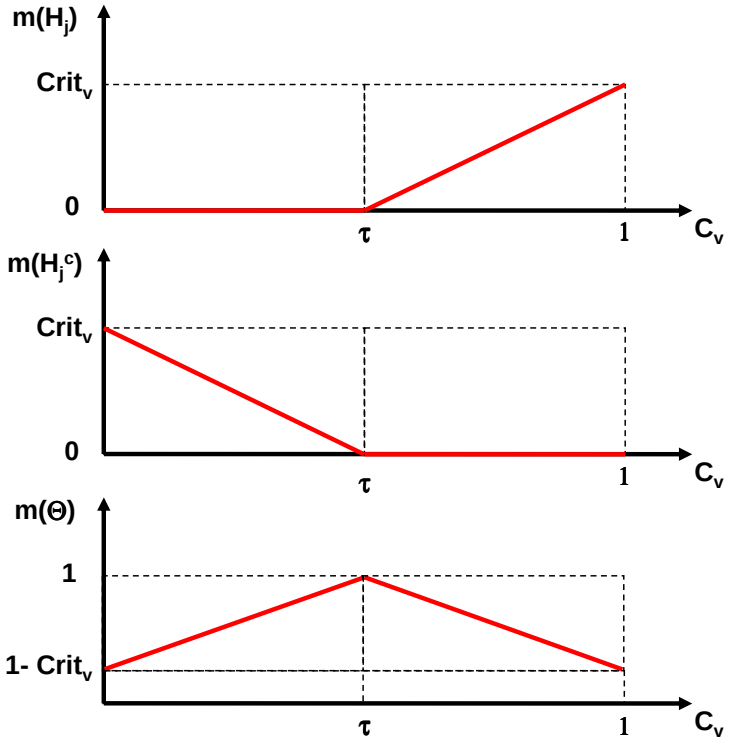



Figure 4.5: Belief Mass Representation

### 4.5.4 Navigation Reliability Quantification

As aforementioned the confidence variable $C_{v_{vis}}$ used for the vision basic belief assignment is directly the Speed Limit Sign Recognition confidence. On the opposite, the confidence variable for the navigation $C_{v_{nav}}$ is more complex as it has to illustrate the reliability of the navigation data. This variable has been determined by going back to the general scheme of a navigation system presented in Chapter.2. Navigation systems are composed of three main elements: the positioning which gives the raw *Vehicle* position, the localization which matches the *Vehicle* position to a digitalized road, and the *Digital Map Database* which gives the information about the road context. These three elements are subject to inaccuracies, which have different origins, but which may lead to false navigation information (e.g. positioning inaccuracies may involve a false localization, thus extracting non relevant information from the digital map). For this reason, it is important to define a confidence indicator $C_{v_{nav}}$ taking account of these factors:

$$C_{v_{nav}} = f(Positioning, Localization, DigitalMapData) \tag{4.23}$$

This variable $C_{v_{nav}}$ consequently represents at the same time the relevancy of the sensors used for positioning ($GPS$, inertial sensors), of the map-matching algorithm and the digital map, thus qualifying completely the reliability of the navigation.

- *The* Vehicle *positioning.* The position is computed using the fusion between the $GPS$ receiver (which gives the longitude and latitude coordinates of the current *Vehicle* position) and odometric and internal sensors. The $GPS$ is mainly used to correct the biases of the dead-reckoning sensors. It is well known that the positioning given by a $GPS$ is not strongly accurate as it is subject to several errors. Most of them are related to the satellite

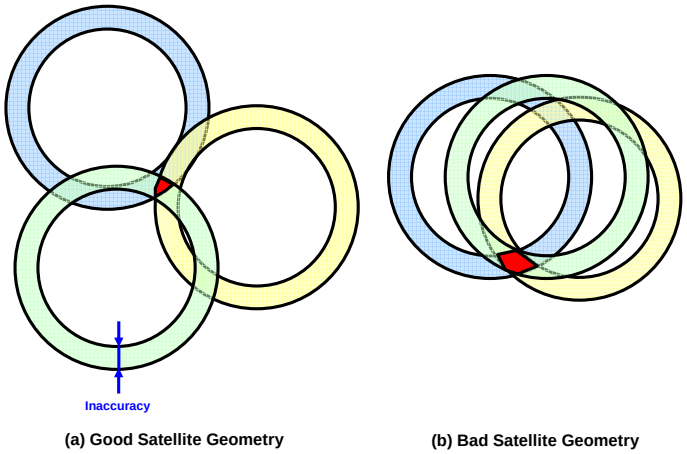



Figure 4.6: Influence of Satellites Geometry

signal transmission (ionospheric deviation, tropospheric deviation, multi-path, etc.) but one of the most important element for $GPS$ positioning is the satellite geometry. Consider the examples presented in Fig.4.6: each satellite signal processing results in *Vehicle*'s coordinates defined regarding a certain accuracy, implying the *Vehicle*'s position to be located in a ring for each satellite. As four satellites are required to determine the *Vehicle* position (three for the coordinates and one for the time), the latter is included in the area defined by the intersection of the satellite rings (red areas in Fig.4.6). Note that the figure only considers three satellites in two dimensions for readability. The comparison of case *(a)* and case *(b)*, clearly reveals that the satellite constellation configuration (location of the satellites) has an important role in the positioning. Indeed, in case *(a)* the intersection area is smaller than the area in *(b)*. Consequently, the position accuracy of case *(a)* is higher than the positioning accuracy of case *(b)*.

This satellite geometry is known as the Geometric Dilution Of Precision ($GDOP$) defined by [Langley, 1999] as:

$$GDOP = \sqrt{\frac{\sigma_E^2 + \sigma_N^2 + \sigma_U^2 + \sigma_T^2}{\sigma}} \tag{4.24}$$

with $\sigma_E$, $\sigma_N$ and $\sigma_U$ the variances respectively in the East, North and up components of the $GPS$ receiver estimates. $\sigma_T$ is the variance of the $GPS$ receiver clock offset estimate and $\sigma$ is the standard deviation of the pseudorange measurement error plus the residual model errors.

In the current context, the navigation system gives a two-dimensional information rather than a 3D position, so the Horizontal Dilution Of Precision ($HDOP$) is considered. This information is relevant of the positioning quality and is defined by:

$$HDOP = \sqrt{\frac{\sigma_E^2 + \sigma_N^2}{\sigma}} \tag{4.25}$$

The lower the $HDOP$ is, the higher the positioning accuracy is.

- *The* Vehicle *localization.* The purpose of this process is to locate the *Vehicle* in the *Digital Map Database* by performing a map-matching of the *Vehicle*'s position. This step can be achieved using different geometrical techniques, topographical techniques, etc. (see [Daniel, 2007] for a description of these different techniques). In the used navigation system, the localization of the *Vehicle* is done by the selection of the most probable candidate. In fact, for each *Vehicle* position, a set of possible *Vehicle* locations are determined regarding to the road context around the *Vehicle* position. Then, using an algorithm which considers the information provided by several sensors (speed, inertial, etc.), a probability is calculated for each candidate. The candidate with the best probability is then selected and considered as the *Vehicle* location. As it is a proprietary navigation system, details about the map-matching algorithm are not available. Nevertheless, the probability of each candidate is available and is used here to define the reliability of the localization steps. For specific cases such as in-city networks, the localization is subject to errors, usually occurring when the probabilities of the different candidate are average and close to each other. However, this information, called Most Likely Candidate Probability ($MLCP$), is relevant of the quality of the localization. Finally, as for the $HDOP$, the lower the probability is, the higher the localization accuracy is.

- *The digital map database accuracy.* As mentioned in Section.2.3.2, the digital map is an approximation of the reality and is subject to inaccuracies. In addition, it is important to note that the representation of the road network is not homogeneous. Indeed, regarding to the importance of the road, the digitalization accuracy is different: an European highway has more points and more attributes than a communal road. In the used *Digital Map Database*, a specific attribute related to the accuracy of the representation is available. This attribute named *ADASAttribute* denotes the quality of the road representation and is active or inactive regarding the considered road. Note that it was previously used in the combination of the navigation criteria as $C_2$.

In summary, the three elements are describing the quality of the positioning, the localization and the *Digital Map Database* and compose the proposed navigation confidence variable such as:

$$C_{v_{nav}} = \left(1 - \left(\frac{HDOP}{HDOP_{\max}}\right)\right) \cdot \left(1 - \left(\frac{MLCP}{MLCP_{\max}}\right)\right) \cdot ADASAttribute \tag{4.26}$$

$HDOP_{\max}$ and $MLCP_{\max}$ have been determined empirically. It can be remarked in Fig.4.7 that this confidence variable is restrictive as it multiplies three values which are upper-bounded by one. Note that this figure presents two surfaces which are defined by the validation or the non validation of the *ADAS Attribute*.

### 4.5.5 Vision Mass Definition

Initially, (cf. [Bradai, 2007]), the vision masses were directly defined using the information provided by the $SLSR$ by taking account of a Forgiveness Factor ($FF$). Indeed, a speed sign which has been detected 5 minutes ago may not be relevant anymore regarding to the road context. Due to this factor, the confidence in the vision information decreases with time (reducing of 0.1 each $30s$). The ignorance increases in the same proportion to provide the normalization to one. The $SLSR$ is a black box which only returns the most probable speed with its probability. The vision mass then corresponds to the confidence of the image processing algorithm in the detected sign ($SLSR_{confidence}$) and the normalization to one is done by setting the ignorance value to the adapted value.

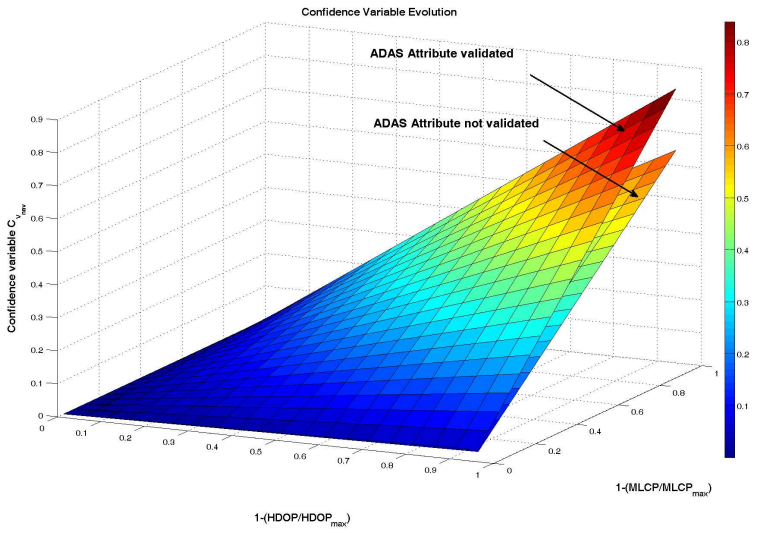



Figure 4.7: Confidence Variable Representation

If, the information obtained form the *SLSR* has not changed, the vision *bba* has been re-defined in the current approach. This step is now processed using the *Rombaut/Gruyer* model, as the navigation information. In the present approach, the vision confidence variable $C_{v_{vis}}$ corresponds to the confidence in the detected speed sign returned by the *SLSR*. Then, by considering that the vision information has a minimal ignorance of 0.1, the belief masses are defined by (4.27) with $m_v(H)$, $m_v(H^c)$ and $m_v(\Theta)$ the masses on the vision speed, the complementary vision speed and the ignorance respectively.

$$\begin{aligned}
m_v(H_v) &= \begin{cases} 0 & (C_{v_{vis}} - FF) \in [0, \tau] \\ \left(\frac{0.9}{1-\tau}\right)(C_{v_{vis}} - FF) - \frac{0.9\tau}{1-\tau} & (C_{v_{vis}} - FF) \in [\tau, 1] \end{cases} \\
m_v(H_v^c) &= \begin{cases} -\frac{0.9}{\tau}(C_{v_{vis}} - FF) + 0.9 & (C_{v_{vis}} - FF) \in [0, \tau] \\ 0 & (C_{v_{vis}} - FF) \in [\tau, 1] \end{cases} \\
m_v(\Theta) &= \begin{cases} \frac{0.9}{\tau}(C_{v_{vis}} - FF) + 0.1 & (C_{v_{vis}} - FF) \in [0, \tau] \\ -\left(\frac{0.9}{1-\tau}\right)(C_{v_{vis}} - FF) + \frac{1-(0.1\tau)}{1-\tau} & (C_{v_{vis}} - FF) \in [\tau, 1] \end{cases}
\end{aligned} \tag{4.27}$$

## 4.6 Speed Limit Definition by Multi-level Fusion

As the selected approach is based on a 2 steps fusion process, a section is dedicated to each level: the multi-criterion and the multi-sensor fusion.

### 4.6.1 Multi-criterion Fusion

Initially, the navigation information was provided by a weighted sum of the criteria masses for several focal elements. These focal elements were determined empirically through the comparison of the speed data stored in the database to the real legal speed limit known for the given driving area. As presented in Table.D.1, each navigation speed refers to at least two focal speeds (except $60km.h^{-1}$ and the *unlimited*). Recent tests allowed a completion of this table - addition to the original table from [Bradai, 2007] are marked in green and removals are marked in red. The

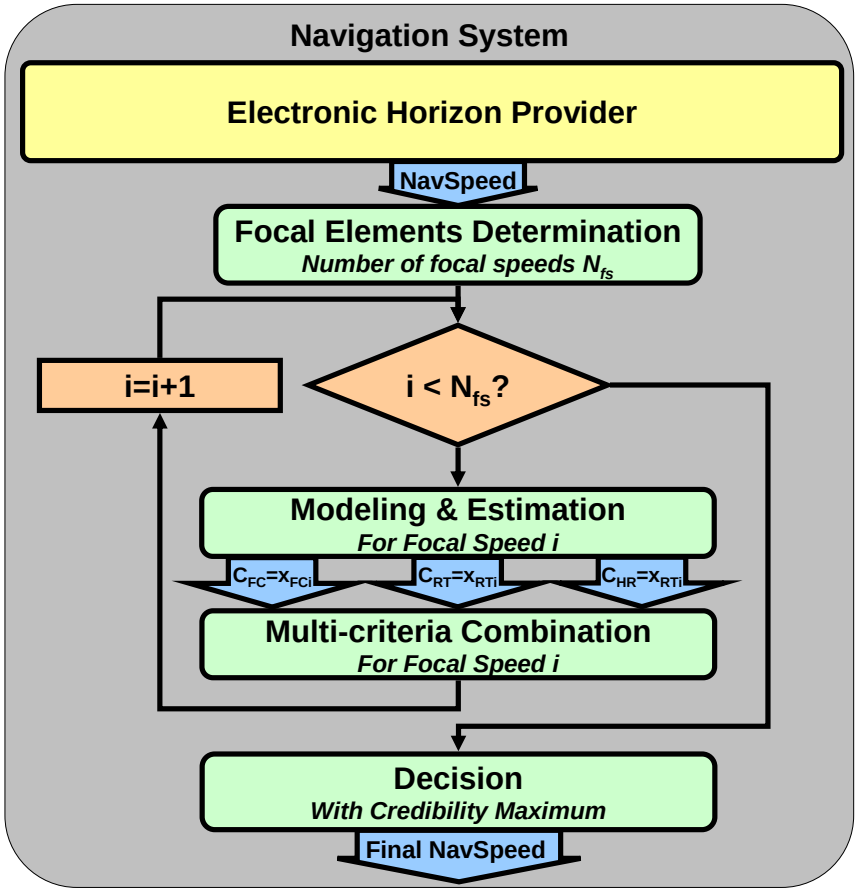



Figure 4.8: Multi-criterion Fusion Scheme

consideration of these focal elements is used to take account of the known errors of the *Digital Map Database*. However, this solution has a limited impact as its exhaustiveness can not be granted. That is why a multi-criterion fusion approach has been considered. By detecting the *Digital Map Database* errors, it provides an information of better quality for the multi-sensor fusion. This is processed through the combination of the information provided by the criteria for the current navigation speed limit based on the basic belief assignment described in Fig.4.5[24]. However, it is important to note that the multi-criterion fusion is done for each focal element as presented in Fig.4.8. Indeed, as shown by this figure, the multi-criterion fusion first determines the focal speeds which are related to the speed limit extracted from the *Digital Map Database*. Then, the criteria *bba* for the first of these focal speeds are defined serially and finally the multi-criterion combination is performed. This step results in the generation of a belief mass set characterizing the considered focal speed. This cycle is then repeated for each focal speed so that a belief mass set is available for each of them. Finally, the decision step selects the speed with the maximum of Credibility. The resulting speed with its set of mass is then considered as the navigation information which will be fused with the vision information during the multi-sensor fusion.

This approach is very interesting as focal speeds represent an important information about the

[24]Remind that here, the criteria are considered as described by independent and specialized sources of information.

possible value of the speed limit in case of erroneous/contradictory data stored in the *Digital Map Database*. For example, lets consider the example provided in Section.4.4: the navigation speed is $50km.h^{-1}$, $C_{FC} = NonValidated$, $C_{RT} = Highway$, $C_C = OutCity$, $C_I = NonValidated$, $C_{HR} = NonValidated$. These criteria configuration describes a navigation information problem as the vehicle is said to drive at $50km.h^{-1}$ on a Highway also with a Functional Class non-validated. The multi-criterion combination for $50km.h^{-1}$ generates a small belief in this speed, however, if this approach helps to detect the error, this lone information does not help to determine the correct navigation speed. The calculation of the belief mass over the different focal speeds then helps to define the most suited one for the multi-sensor fusion. Note that another solution could be to apply the fusion for all the speeds of the discernment frame. However, this solution is eager in calculation time, and may not bring strong enhancements considering that the potential speeds are already stored in the focal elements table.

To clearly describe the multi-criterion fusion, a reminder of the notations is proposed. Remind that the multi-criterion fusion presented in this section is based on the works performed by [Royère, 2002] and by [Mourllion, 2006].

Let $\Theta$ be the frame of discernment containing the different hypotheses (here the different speeds) such that:

$$\Theta = \{H_1, H_2, ..., H_k\} \tag{4.28}$$

with $k$ the number of hypotheses.

Due to the cyclic approach for the focal elements *bba*, the multi-criterion fusion is based on a succession of combinations of sources which gives information about the same speed. The frame of discernment is then specialized to $\Theta = \{H_i, H_i^c\}$ for each combination step, so defined over the following referential subset $2^\Theta$:

$$2^\Theta = \{\emptyset, H_i, H_i^c, \Theta\} \tag{4.29}$$

The open-world assumption is made here so allows conflict mass $\emptyset \geq 0$. Then let $m_{j,i}(H_i)$ be the mass of the criterion $j$ over the speed $i$. The multi-criterion combination is obtained using *Dempster*'s conjunctive operator for the $l$ criteria such that:

$$m_{1...l,i} = m_{1,i} \oplus m_{2,i} \oplus \ldots \oplus m_{l,i} \tag{4.30}$$

The used basic belief assignment model (based on the *Rombaut/Gruyer* model) involves a non-generation of conflict ($m_{j,i}(\emptyset) = 0$). Considering this property, the multi-criterion combination is then defined for each hypothesis $H_j$ by (4.30) and described by Table.4.6 for the $k$ hypotheses over the $l$ criteria. For each cycle, thus for each focal speed, a combination of one column of Table.4.6 is done.

Finally, a generalized form for the combination of $l$ specialized sources on the same hypothesis $H_j$ is given in [Royère, 2002] ($p.37$) as following:

Table 4.6: Multi-criterion Combination Table

| | $H_1$ | $H_2$ | ... | $H_k$ |
|---|---|---|---|---|
| $C_1$ | $m_{1,1}(H_1)$<br>$m_{1,1}(H_1^c)$<br>$m_{1,1}(\Theta)$ | $m_{1,2}(H_2)$<br>$m_{1,2}(H_2^c)$<br>$m_{1,2}(\Theta)$ | ... | $m_{1,k}(H_k)$<br>$m_{1,k}(H_k^c)$<br>$m_{1,k}(\Theta)$ |
| $C_2$ | $m_{2,1}(H_1)$<br>$m_{2,1}(H_1^c)$<br>$m_{2,1}(\Theta)$ | $m_{2,2}(H_2)$<br>$m_{2,2}(H_2^c)$<br>$m_{2,2}(\Theta)$ | ... | $m_{2,k}(H_k)$<br>$m_{2,k}(H_k^c)$<br>$m_{2,k}(\Theta)$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\ddots$ | $\vdots$ |
| $C_l$ | $m_{l,1}(H_1)$<br>$m_{l,1}(H_1^c)$<br>$m_{l,1}(\Theta)$ | $m_{l,2}(H_2)$<br>$m_{l,2}(H_2^c)$<br>$m_{l,2}(\Theta)$ | ... | $m_{l,k}(H_k)$<br>$m_{l,k}(H_k^c)$<br>$m_{l,k}(\Theta)$ |
| Combination Results | $m_{1...l,1}(H_1)$<br>$m_{1...l,1}(H_1^c)$<br>$m_{1...l,1}(\Theta)$<br>$m_{1...l,1}(\emptyset)$ | $m_{1...l,2}(H_2)$<br>$m_{1...l,2}(H_2^c)$<br>$m_{1...l,2}(\Theta)$<br>$m_{1...l,2}(\emptyset)$ | ... | $m_{1...l,k}(H_k)$<br>$m_{1...l,k}(H_k^c)$<br>$m_{1...l,k}(\Theta)$<br>$m_{1...l,k}(\emptyset)$ |

$$
\begin{aligned}
m_{1...l,i}(H_i) &= \prod_{j=1}^{l}(1-m_{j,i}(H_i^c)) - \prod_{j=1}^{l} m_{j,i}(\Theta) \\
&= \prod_{j=1}^{l}(m_{j,i}(H_i)+m_{j,i}(\Theta)) - \prod_{j=1}^{l} m_{j,i}(\Theta) \\
m_{1...l,i}(H_i^c) &= \prod_{j=1}^{l}(1-m_{j,i}(H_i)) - \prod_{j=1}^{l} m_{j,i}(\Theta) \\
&= \prod_{j=1}^{l}(m_{j,i}(H_i^c)+m_{j,i}(\Theta)) - \prod_{j=1}^{l} m_{j,i}(\Theta) \\
m_{1...l,i}(\Theta) &= \prod_{j=1}^{l} m_{j,i}(\Theta) \\
m_{1...l,i}(\emptyset) &= 1 - \prod_{j=1}^{l}(1-m_{j,i}(H_i)) - \prod_{j=1}^{l}(1-m_{j,i}(H_i^c)) + \prod_{j=1}^{l} m_{j,i}(\Theta) = 0
\end{aligned}
\tag{4.31}
$$

As presented in Fig.4.8, the combination is processed sequentially for all the focal speeds related to the extracted navigation speed. The results of the multi-criterion combination is consequently composed of several sets $\{m_{1...l,i}(H_i), m_{1...l,i}(H_i^c), m_{1...l,i}(\Theta)\}$, with $i = 1, 2, ..., N_{fs}$ with $N_{fs}$ the number of focal speeds. The selection of the final navigation speed retained for the fusion with the vision speed is based on the maximum of Belief criterion:

$$
m_n(H_n) = \max_{1\le i\le N_{fs}} Bel(H_i) = \max_{1\le i\le N_{fs}} m_{1...l,i}(H_i) \tag{4.32}
$$

with $m_n(H_n)$ the belief in the final navigation speed $H_n$.

### 4.6.2 Multi-sensor Fusion

This combination constitutes the second step of the fusion scheme presented in Fig.4.3: fusing the information coming from the sensors. Contrary to the multi-criterion fusion, the combination

Table 4.7: Multi-sensor Combination Table

| | Multi-Sensor Combination | | | |
|---|---|---|---|---|
| | $S_1$ | $S_2$ | ... | $S_p$ |
| | $m_1(H_1), m_1(H_1^c), m_1(\Theta)$ | $m_2(H_2), m_2(H_2^c), m_2(\Theta)$ | ... | $m_p(H_p), m_p(H_p^c), m_p(\Theta)$ |
| Combination Results | $m_{1...p}(\emptyset)$<br>$m_{1...p}(H_1)$<br>$m_{1...p}(H_2)$<br>$\vdots$<br>$m_{1...p}(H_p)$<br>$m_{1...p}(H_1 \cup H_2)$<br>$\vdots$<br>$m_{1...p}(H_j \cup H_l)$<br>$\vdots$<br>$m_{1...p}(H_1 \cup \ldots \cup H_l)$<br>$\vdots$<br>$m_{1...p}(\Theta)$ | | | |

is here applied over sensors which may have different points of view, so different speed values. In addition, each sensor is specialized on a speed which can be any speed of the discernment frame presented in (4.33).

$$\Theta = \{H_1, H_2, ..., H_k\} \tag{4.33}$$

The multi-sensor combination is obtained using *Dempster*'s operator for the different sensors such that:

$$m_{1...p,i} = m_{1,i} \oplus m_{2,i} \oplus \ldots \oplus m_{p,i} \tag{4.34}$$

with $p$ the number of sensors.

As sensors are based on the same basic belief assignment model, they consequently are characterized by masses on the same triplet: mass on the speed $m_{j,i}(H_i)$, on the opposite speed $m_{j,i}(H_i^c)$ and on the ignorance $m_{j,i}(\Theta)$. For the sake of clarity, the notations are simplified so that a sensor $j$ gives mass over its speed $j$ such as $m_j(H_j)$ with $H_j \subseteq \Theta$ and the multi-sensor combination over a speed $j$ is noted $m_{1...p}(H_j)$. Considering these elements, the result of the multi-sensor combination is presented in Table.4.7 for $p$ sensors [Royère, 2002].

As for the multi-criterion combination, the results presented in Table.4.7 can be generalized in the following way:

$$
\begin{aligned}
m_{1\ldots p}(H_j) &= m_j(H_j) \prod_{\substack{a=1 \\ a\neq j}}^{p} (1 - m_a(H_a)) + m_j(\Theta) \prod_{\substack{a=1 \\ a\neq j}}^{p} (m_a(H_a^c)) \\
m_{1\ldots p}(H_j \cup H_l) &= m_j(\Theta)\, m_l(\Theta) \prod_{\substack{a=1 \\ a\neq j \\ a\neq l}}^{p} m_a(H_a^c)
\end{aligned}
$$

and for union combinations of 2 to p-1 hypotheses:

$$
\begin{aligned}
m_{1\ldots p}(H_j \cup \ldots \cup H_l) &= m_j(\Theta) \ldots m_l(\Theta) \prod_{\substack{a=1 \\ a\neq j \\ \ldots\ldots \\ a\neq l}}^{p} m_a(H_a^c) \\
m_{1\ldots p}\left(H_j^c\right) &= m_j\left(H_j^c\right) \prod_{\substack{a=1 \\ a\neq j}}^{p} m_a(\Theta) \\
m_{1\ldots p}(\Theta) &= \prod_{a=1}^{p} m_a(\Theta) \\
m_{1\ldots p}(\emptyset) &= 1 - \left[ \prod_{a=1}^{p} (1 - m_a(H_a)) + \sum_{a=1}^{p} m_a(H_a) \prod_{\substack{b=1 \\ b\neq a}}^{p} (1 - m_b(H_b)) - \prod_{a=1}^{p} m_a(H_a^c) \right]
\end{aligned}
\tag{4.35}
$$

It can be noticed that $m_{1\ldots p}\left(H_j^c\right)$ corresponds to the mass on union combinations of $p-1$ hypotheses.

The current equations are applied to a *SLA* combining information from a camera and a navigation system which gives only information about one speed. Consequently $p$ is equal to 2, which implies $k$ to be also equal to 2. The frame of discernment is then considered to be only composed of the sensor speeds such as:

$$\Theta = \{H_v, H_n\} \tag{4.36}$$

which is defined on the following referential subset:

$$2^{\Theta} = \{\emptyset, H_v, H_n, \Theta\} \tag{4.37}$$

with $H_v$ the vision speed and $H_n$ the navigation speed[25]. This greatly simplifies the multi-sensor fusion equations presented in (4.35) into:

$$
\begin{aligned}
m_{vn}(H_v) &= m_v(H_v)(1 - m_n(H_n)) + m_v(\Theta)\, m_n(H_n^c) \\
m_{vn}(H_n) &= m_n(H_n)(1 - m_v(H_v)) + m_n(\Theta)\, m_v(H_v^c) \\
m_{vn}(\Theta) &= m_v(\Theta)\, m_n(\Theta) \\
m_{vn}(\emptyset) &= m_v(H_v)\, m_n(H_n) + m_v(H_v^c)\, m_n(H_n^c)
\end{aligned}
\tag{4.38}
$$

Note that, as the discernment frame $\Theta$ is supposed to be exhaustive and as there are only two sources, $m_{vn}(H_v^c)$ and $m_{vn}(H_n^c)$ are directly integrated into $m_{vn}(H_n)$ and $m_{vn}(H_v)$.

Finally, the specific case in which both sensors are specialized on the same speed $H_i$, thus based on the *discernment frame* $\Theta = \{H_i, H_i^c\}$, results in the consideration of the combination

[25] As there are only two possible speeds, the union of these speeds obviously corresponds to the ignorance ($H_v \cup H_n = \Theta$).

rules used for specialized sources on the same hypothesis. (4.38) is then transformed into (4.39) which corresponds to the application of (4.31) for 2 sources.

$$\begin{array}{ll} m_{vn}(H) & = m_v(H) \cdot (m_n(\Theta) + m_n(H)) + m_v(\Theta) \cdot m_n(H) \\ m_{vn}(H^c) & = m_v(H^c) \cdot (m_n(\Theta) + m_n(H^c)) + m_v(\Theta) \cdot m_n(H^c) \\ m_{vn}(\Theta) & = m_v(\Theta) \cdot m_n(\Theta) \\ m_{vn}(\emptyset) & = m_v(H) \cdot m_n(H^c) + m_v(H^c) \cdot m_n(H) = 0 \end{array} \tag{4.39}$$

## 4.7 Conflict Management and Final Decision

Contrary to the multi-criterion combination, the sources may give information about different speeds. These cases involve the generation of conflict which has to be processed before the *Decision*. Initially, the conflict was directly redistributed using the *Dempster* strategy, so redistributed proportionally on all the referential subsets elements (cf. (4.13)). In the current Speed Limit Assistant, a new strategy consisting in the comparison of a raw and conflict redistributed fusion, has been adopted:

- The first information is referred to the direct result of the combination without conflict redistribution. Indeed, the conflict can be interpreted as an information related to the unreliability of the sources or to the non-exhaustiveness of the discernment frame. It will be shown, in the tests presented in Section.5.5, that this case usually refers to errors of the $SLSR$ (detection of signs which are related to another class of vehicles, etc.). Nevertheless, it may also correspond to a conflict between sensor information, thus generating a very high conflict. In these particular cases, for safety reasons, no decision can be taken. Indeed, we cannot take the risk to select a speed on which the belief is not high enough. The first speed information provided to the *Driver* will then be the speed having the maximum of belief considering a minimum threshold. This threshold, fixed to 0.5, avoids the selection of a speed for which the system has not enough confidence. If both speeds have a belief which is under the threshold, the retained speed is $undefined$ with a belief of 1.0[26]:

$$\begin{array}{ll} H_{final} & = \left\{ arg \max\limits_{1 \leq j \leq k} Bel(H_j) \geq 0.5, undefined \right\} \\ & = \{ arg \max \{ Bel(H_v) \geq 0.5, Bel(H_n) \geq 0.5 \}, undefined \} \end{array} \tag{4.40}$$

- The second information provided to the *Driver* refers to the technique used until now in the Speed Limit Assistant: the redistribution of the conflict. However, as mentioned previously, the *Dempster* redistribution is not adapted to the current situation as it redistributes the conflict proportionally on all the masses. Consequently, this approach redistributes the conflict which may be generated by only a few sensors, over all the masses (even the ignorance). In the current approach, *Florea*'s redistribution operator [Florea et al., 2006], named *Proportional Conflict Redistribution* ($PCR$), has been retained. Indeed, it redistributes the conflict proportionally on the sources generating it. This operator is defined by:

$$m_{PCR}(H_j) = m_\wedge(H_j) + \sum_{\substack{H_a \in \Theta \setminus \{H_j\} \\ H_j \cap H_a = \emptyset}} \left[ \frac{m_j(H_j)^2 m_a(H_a)}{m_j(H_j) + m_a(H_a)} + \frac{m_a(H_j)^2 m_j(H_a)}{m_a(H_j) + m_j(H_a)} \right] \tag{4.41}$$

[26] In this case, the Speed Limit Assistant is in complete ignorance

with $m_{\wedge}(H_j)$ the mass on hypothesis $H_j$ after the conjunctive combination, and $m_{PCR}(H_j)$ the mass on hypothesis $H_j$ after the conflict redistribution. The application of the $PCR$ obviously involves a closed-world consideration ($m_{\emptyset} = 0$) and higher masses on the sensors speeds. Considering this point, the *Decision* step is based on the maximum of Belief over the navigation and vision masses after the $PCR$:

$$H_{final} = arg \max_{1 \leq j \leq k} Bel\left(H_j\right) = arg \max \left\{Bel\left(H_v\right), Bel\left(H_n\right)\right\} \tag{4.42}$$

This information may be contradictory to the statement of the preceding point as it tries to give an information to the *Driver* even if the conflict is very high. Nevertheless, this information, as it is based on the redistribution of the conflict on its generators, may also be an interesting information about the current speed limit. Indeed, sources which are generating a high conflict have usually strong beliefs in their proposition. These beliefs are greatly reduced after the combination as integrated in the conflict. The use of this redistribution operator can therefore involve a re-apparition of the strong belief which are originating the conflict while preserving the other information obtained from the combination (ignorance, etc.).

The information provided to the *Driver* is consequently of two forms: a pragmatic and safe information based on the results of the combination step (which considers the conflict as an information) and an indication about the most probable speed through the redistribution of the conflict on the sensor speeds. The comparison of these two information is provided in Section.5.5.

## 4.8 Conclusion

In this Chapter the description of this PhD $SLA$ has been presented. The latter is based on a two-level data fusion consisting in a local processing of the sensor data and in a multi-sensor fusion. Compared to conventional approaches such as [Bradai, 2007] or [Puthon et al., 2010] three main improvements have to be retained done:

- The first one is related to the belief mass modeling which has been completely refined regarding the previous approach. The estimation is now processed based on the *Rombaut/Gruyer* model which considers sources to be independent and specialized on a specific hypothesis. This implies sources to give an opinion only about the triplet: *"It is this hypothesis"*, *"It is not this hypothesis"* or *"I do not know"*. In addition, this model is based on the consideration that a source cannot give contradictory information, i.e. cannot say *"It is this hypothesis"* and *"It is not this hypothesis"* at the same time. The use of this model avoids the generation of conflict in the multi-criterion fusion step in which sources are giving their opinion on the same hypothesis.

- The second major improvement refers to the definition of the confidence variable used during the belief mass estimation. This variable is a numerical representation of the reliability which can be imputed to the considered sensor, i.e. is a representation of the sensor level of inaccuracy. Here the main contribution lies in the consideration of all the navigation system inaccuracies for the determination of the navigation confidence variable. Indeed, it is composed of a positioning, localization and *Digital Map Database* reliability indicators respectively represented by the $HDOP$, the $MLCP$ and the $ADASAttribute$. This is directly related to the redefinition of the criteria role regarding navigation information.

Indeed, it has been shown that criteria can be classified into two categories: criteria describing the road context and criteria indicating the navigation system reliability. The role of each criterion has consequently been redefined to best suit their respective nature.

- The last major improvement concerns the selection of the final information which is sent to the *Driver*. Two information is now given to the *Driver*: the raw fusion results and the conflict redistributed fusion results. The first one, gives the results of the fusion without conflict redistribution. This copes with the conventional approaches which were automatically redistributing the conflict using the *Dempster* normalization operator. The first information consequently refers to the speed having the maximum of belief regarding a minimum threshold, thus allows the system to give an *undefined* speed limit. The second information consists in the results obtained after redistributing the conflict using *Florea*'s operator, which redistribute it only over the sources which generate it.

With this Chapter, the description of the approaches and techniques adopted for constraint management and information combination applied in the Navigation-aided *ADAS* is now complete. The next Chapter can consequently focus on the results obtained with these systems.

# Chapter 5

# Experimental Results

**Contents**



## 5.1 Introduction

In Chapter.2, the strategy adopted to apply the constraints management and the information combination to the Navigation-based *ADAS*, has been presented. Chapter.3 and Chapter.4 were then dedicated to the detailed presentation of the unconstrained/constrained trajectory generation and the multi-level data fusion techniques considered in this PhD as well as their respective applications. The present Chapter can consequently be focused on the presentation of the results obtained with such systems and is organized as follows:

- Simulation results of the unconstrained trajectory generation are depicted. They will confirm the expected benefits of the $PCS$ and the limitation of the unconstrained approach which involved the consideration of the *System* and *Environment* constraints during the *Controller* synthesis. Anyway, the application of the unconstrained trajectory generation to the Longitudinal Controller gives satisfactory results.
- The next step is the presentation of the results obtained using the constrained trajectory generation which is also divided into two steps: the presentation of simulations and their application to the Lateral controller. The first one compares the constrained trajectories generated using the different cost criteria. From this comparison, it will emerge that the minimization of the energy provides trajectories of better quality. This is confirmed by the application of the different trajectories to the Lateral Controller. Indeed, during these tests, the measurement estimation of the electric power required by the test *Vehicle* steering motor has been performed, thus helps to determine the energy consumption involved by each trajectory tracking solution.
- Finally a large part of this Chapter is dedicated to the presentation of the results obtained with the multi-level $SLA$. As for both previous approaches, the presentation of these results is divided into two parts: simulation and real-time tests. The simulations are dedicated to the comparison of each multi-level step to the original weighted sum approach presented in [Bradai, 2007], which will show the benefits of the proposed techniques. Finally real-time $SLA$ results, which will confirm its benefits, are shown.

The test *Vehicle* used for these applications is presented in Appendix.A.

## 5.2 Road Model Validation

Since the road model is used to define the location of the points required by the unconstrained trajectory generation and used to define the validity area (so related to the constrained trajectory generation), it is important to validate its computation principle. As mentioned previously, the road model is composed of three continuous curves: the road centerline, the left road boundary curve and the right road boundary curve. Comparison tests were carried out with real road data corresponding to a race track. These data were obtained via topological measurements with a centimeteric accuracy. The goal is now to compare the real road data with the road model defined by Splines. Note that the generated trajectories presented in this section have been computed using centripetal parameter values and based on the shape points of a *Digital Map Database* (black points in Fig.5.1). Fig.5.1 presents the results of the road model estimation along the entire race track. It is difficult to distinguish any difference between the real data and the Spline. This is confirmed by Fig.5.2 top plot which selects a portion of the race track. *Area 1* and *Area 2* correspond to relevant areas which will be used for the presentation of constrained trajectory results in Section.5.4.

In Fig.5.2 bottom plot, the results of the error between the real road centerlane against the estimated road centerlane is presented. The latter shows that the error is close or lower than $50cm$ with a mean value of $29cm$. This confirms that, with only the information of the *Digital Map Database*, accurate results in the road profile reconstruction can be obtained. It can also be noted that the Spline has better results in bends than in straight lines. This can be explained by the points density which is more important in bends. In these situations, the Spline is more constrained, contrary to the straight lines on which the Spline has less points to interpolate, thus less constraints. Nevertheless the error is acceptable.

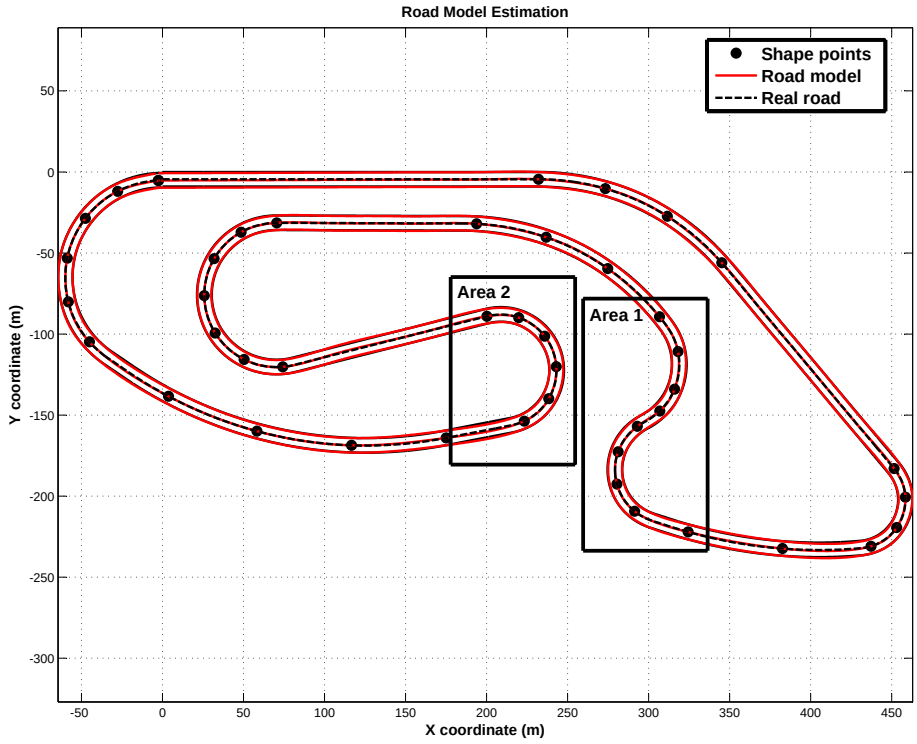



Figure 5.1: Considered Test Track

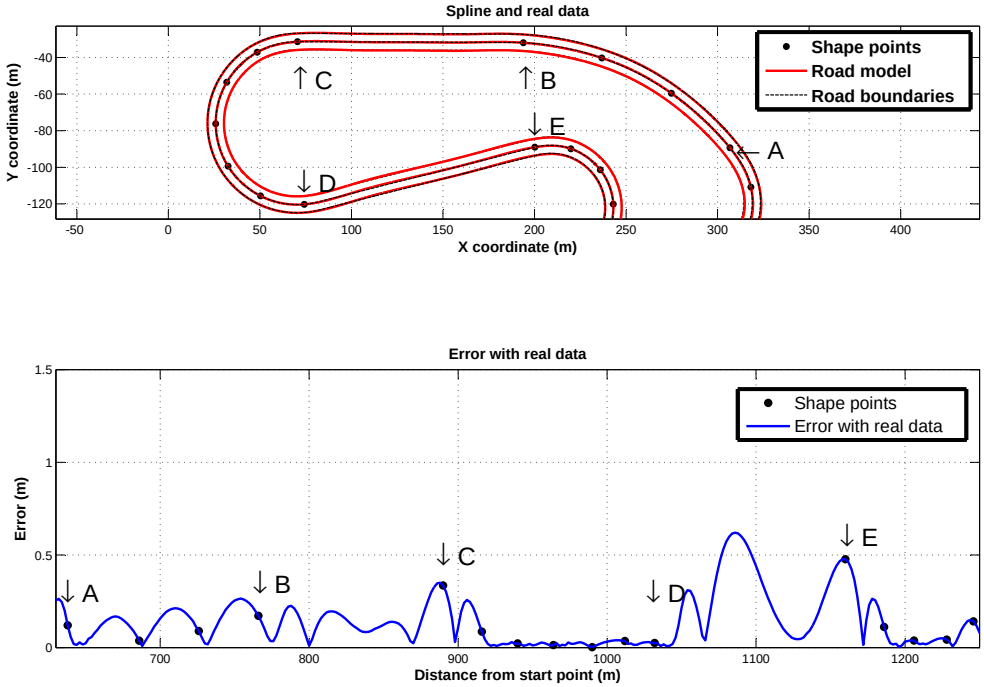



Figure 5.2: Road Model Estimation Focus

## 5.3 Unconstrained Trajectory Generation Results

This section is dedicated to the description of the results obtained using the unconstrained trajectory generation. This includes simulation results and results obtained with the Longitudinal Controller.

### 5.3.1 Simulation Results

As for the determination of the boundary conditions and the parameter values, the validation of the unconstrained trajectory generation has been done on the race track presented in Fig.5.1. From the road model validation tests presented in the previous section, it has been shown that the Spline model has a decimeteric accuracy. The current goal is then to study the Spline properties which are its location in the road lane and its curvature continuity. Fig.5.3 presents a portion of the unconstrained trajectory generated on the entire race track. This trajectory is based on Spline defined with *Special* conditions and parameter values which depend on the square root of the distance between the interpolated points (cf. (5.1) and (5.2)).

$$\begin{cases} \ddot{f}(t_0) = \begin{cases} 2\frac{\left(\frac{X_2-X_1}{h_1}\right)-\left(\frac{X_1-X_0}{h_0}\right)}{h_1+h_0} \\ 2\frac{\left(\frac{Y_2-Y_1}{h_1}\right)-\left(\frac{Y_1-Y_0}{h_0}\right)}{h_1+h_0} \end{cases} \\ \ddot{f}(t_n) = \begin{cases} 2\frac{\left(\frac{X_n-X_{n-1}}{h_{n-1}}\right)-\left(\frac{X_{n-1}-X_{n-2}}{h_{n-2}}\right)}{h_{n-1}+h_{n-2}} \\ 2\frac{\left(\frac{Y_n-Y_{n-1}}{h_{n-1}}\right)-\left(\frac{Y_{n-1}-Y_{n-2}}{h_{n-2}}\right)}{h_{n-1}+h_{n-2}} \end{cases} \end{cases} \tag{5.1}$$

$$t_{i+1} = \sum_{i=0}^{n-2} \left( (X_{i+1} - X_i)^2 + (Y_{i+1} - Y_i)^2 \right)^{\frac{1}{4}} \; with \; t_0 = 0 \tag{5.2}$$

The top plot of Fig.5.3 presents the road boundaries (in full black lines) and its middle line (in dotted line) obtained from the architect data. The black points correspond to the database shape points of the test track used to generate the unconstrained trajectory. They also correspond to the points which could be seen on the other plot of the figure. Finally, the generated trajectory is represented in full red line. The shape of the trajectory, located in the right lane of the test track, is similar to the road lines (centerline and boundaries). It is clearly shown that the trajectory never leaves the lane, thus proving the efficiency of the present trajectory generation method. In addition, the correspondence between the road profile and the curvature shape, in the bottom plot, appears clearly. For the first bend (point $A$ to point $B$), the curvature is large and positive. Between points $B$ and $C$, there is a straight line; the curvature returns therefore close to zero. In the second bend (point $C$ to point $D$) the curvature is large and negative. The following smooth bend (point $E$ to point $F$) shows a curvature which is still negative but closer to zero. Finally, as the bend becomes sharper, the curvature increases again. This shows that the curvature follows the road shape correctly. This figure finally shows that the curvature is almost constant for the different bends, especially in the two first ones (between points $A$ and $B$ and between points $C$ and $D$), which is coherent with the real curvature (is black dashed line).

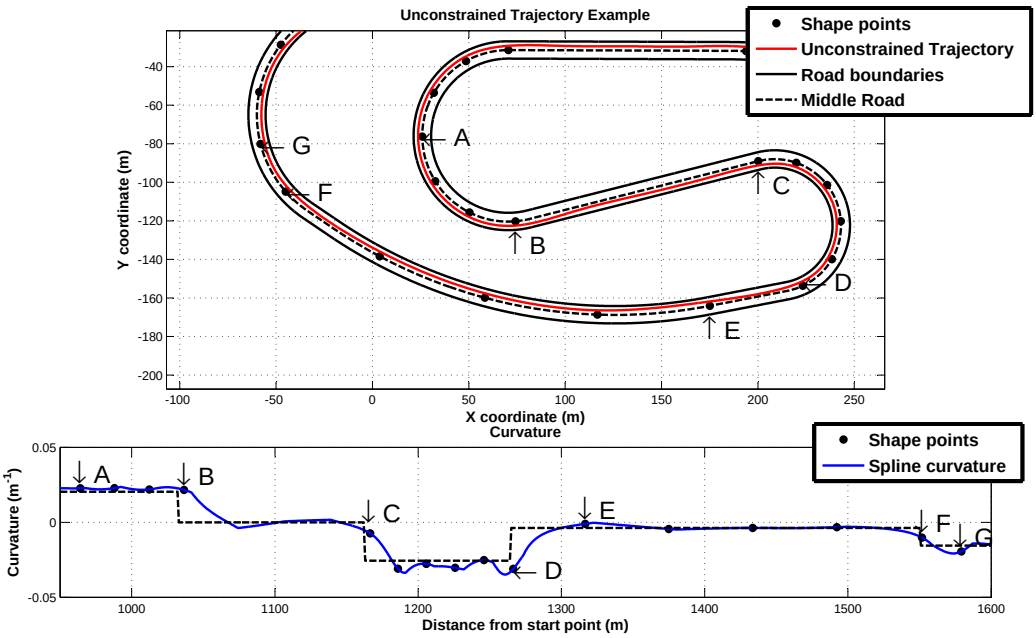



Figure 5.3: Unconstrained Trajectory Example

### 5.3.2 Navigation-based Longitudinal Control Results

#### 5.3.2.1 Speed Profile Generation

As mentioned previously, the Longitudinal Controller is based on the calculation of a limit speed profile defined by the unconstrained trajectory and the reference speed model (cf. (5.3)) which mostly depends on the curvature (cf. (5.4)). The curvature variations therefore involve a variation of the corresponding limit speed profile as presented in Fig.5.4. Indeed, in this figure, it is clear that race track sections which have a large curvature i.e. bends, correspond to small speed limit values contrary to straight lines. Note that in Fig.5.4, a maximum speed limit value of $130km.h^{-1}$ has been chosen as it represents the maximum speed allowed on French roads and as it also avoids infinite speeds in straight lines (since straight line curvature is equal to zero).

$$V_{limit} = \sqrt{\frac{\gamma_{T_{max}}}{\kappa}} \tag{5.3}$$

$$\kappa\left(x(t), y(t), t\right) = \frac{\ddot{y}\left(t\right)\dot{x}\left(t\right) - \ddot{x}\left(t\right)\dot{y}(t)}{\left(\dot{x}^2\left(t\right) + \dot{y}^2\left(t\right)\right)^{\frac{3}{2}}} \tag{5.4}$$

However, this figure also presents the limitations of the unconstrained trajectory coupled to the limit speed profile calculation. Indeed, between the successive right and left bend (cf. Fig.5.5), it can be seen that the speed limit increases rapidly from $40km.h^{-1}$ to $130km.h^{-1}$ and then decreases rapidly from $130km.h^{-1}$ to $40km.h^{-1}$ in only a few meters. This is due to a brief change in the curvature of the generated trajectory. It is obvious that classical vehicles are not able to follow this speed profile. As the unconstrained trajectory generation does not consider this aspect, it has to be managed in the Controller synthesis step as a constraint.

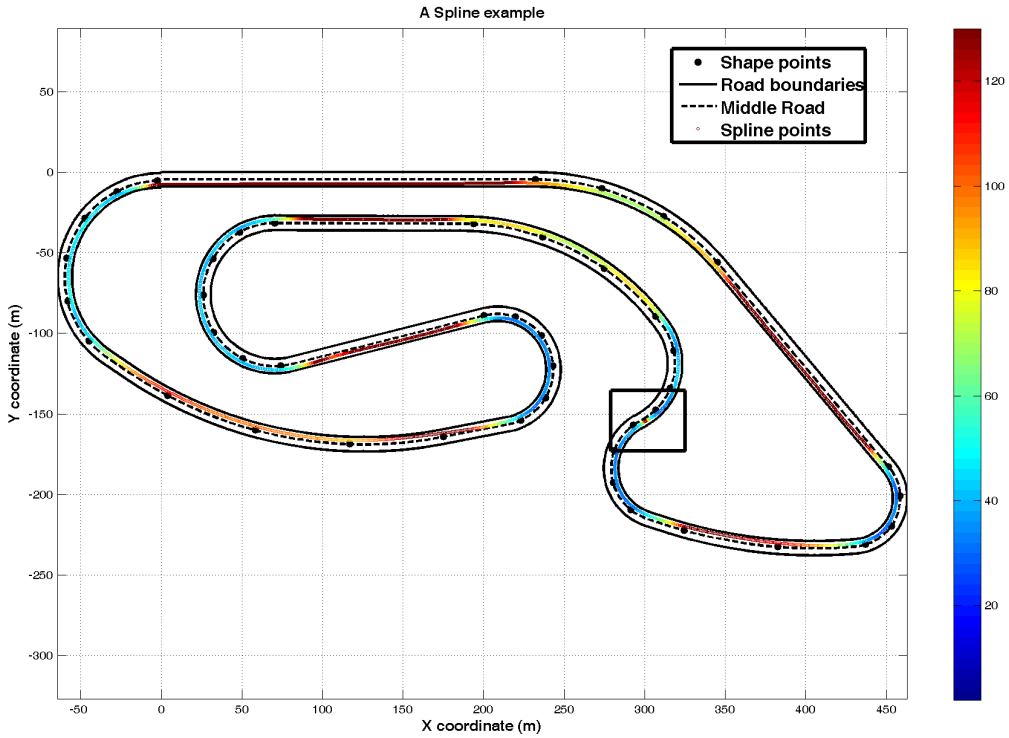



Figure 5.4: Limit Speed Profile

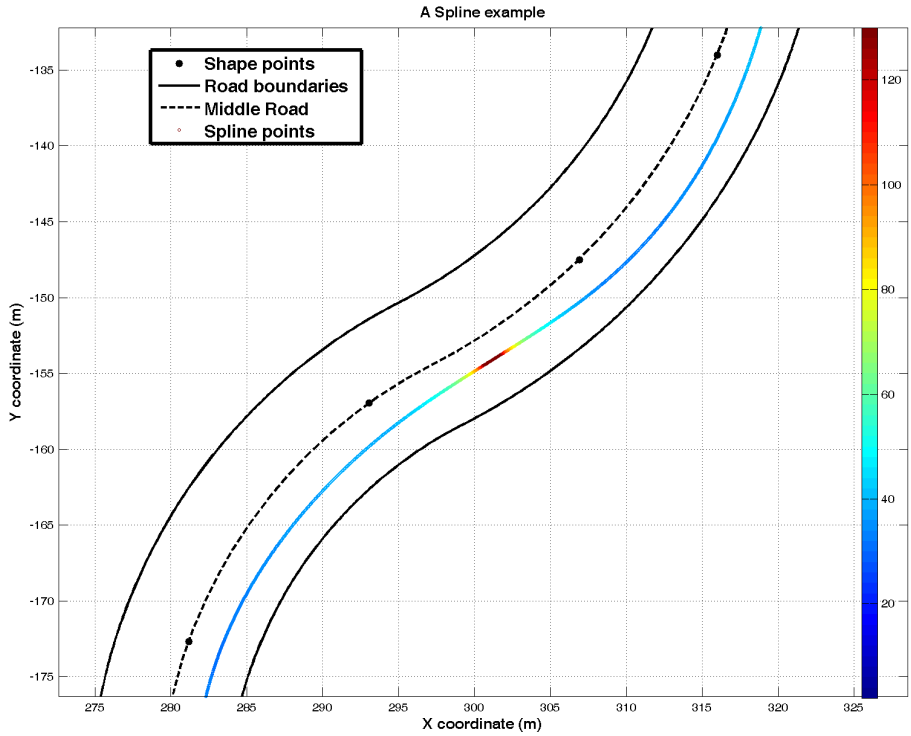



Figure 5.5: Limit Speed Profile Limitations

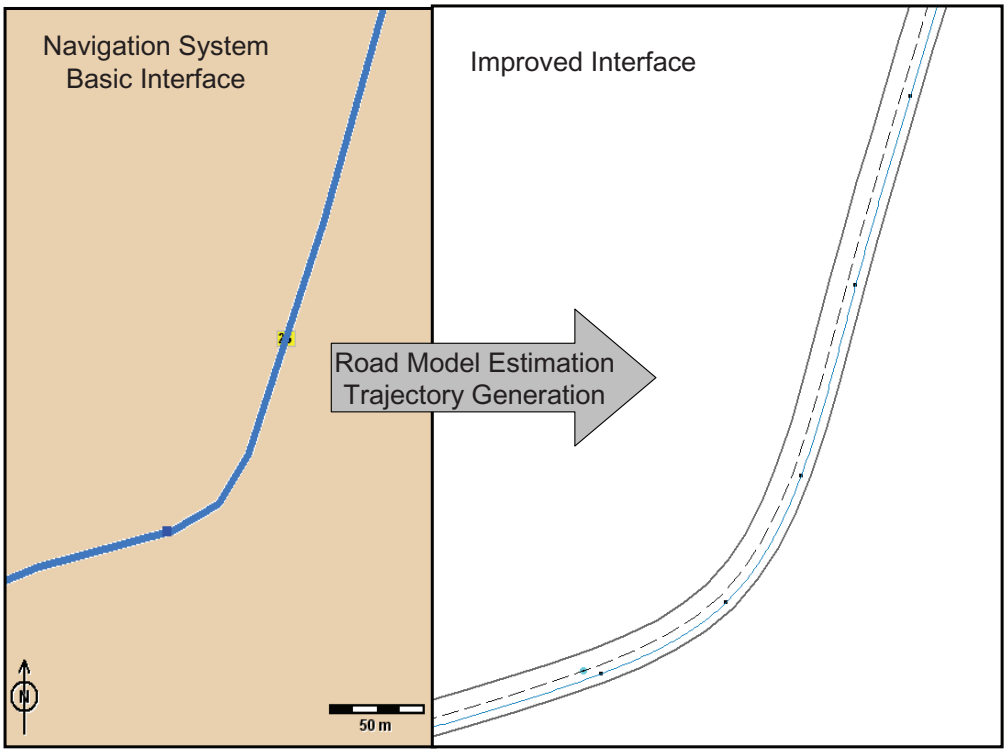



Figure 5.6: Graphical Interface Improvements

#### 5.3.2.2 Real Tests Conditions

The speed profile modification and the different techniques presented in Section.3.4.2 were tested dynamically on open roads with the test car. These tests were carried out in a semi-automatic mode: longitudinal control acted directly on the different actuators. However, the *Driver* was able to take the control of the *Vehicle* back in his hands at any moment.

Fig.5.6 shows a road example with its corresponding navigation system map in the left part, and the calculated road model with the corresponding lane trajectory in the right part. This trajectory is used to determine the limit speed profile. It can be noted that *ADASRP*, as most of the other navigation systems, is based on a graphical interface which only presents data related to the middle of the road. However, this software allows the definition of a secondary graphical interface, which is here presenting the elements referring to the road model calculation and to the unconstrained trajectory generation, so provide a more complete interface.

As for the classical *ADASRP* interface, the developed interface is real-time compliant as for the unconstrained trajectories.

The considered road for the Longitudinal Control tests is shown in Fig.5.7. It corresponds to a succession of a left and a right bend. It can be foreseen that the *Vehicle* will brake before the left bend, then maintain the speed during a short period and then accelerate until reaching the braking location implied by the second bend.

#### 5.3.2.3 Results

The behavior of the Longitudinal Controller, on this road, is presented in Fig.5.8. This figure presents the unconstrained trajectory (black line) used for the calculation of the limit speed profile

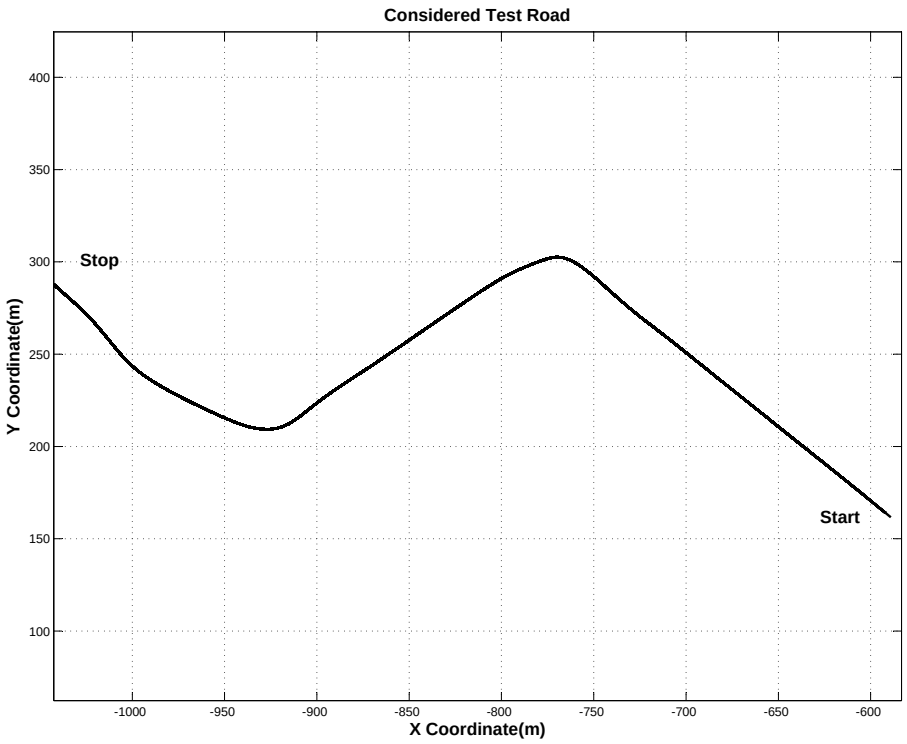



Figure 5.7: Considered Test Road

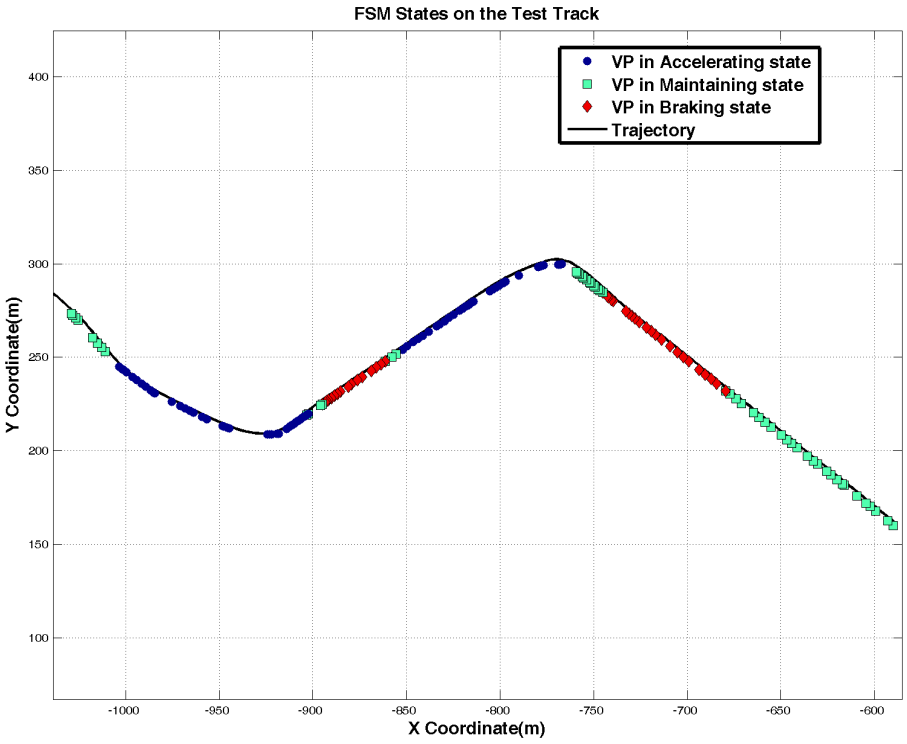



Figure 5.8: Longitudinal Control Behavior

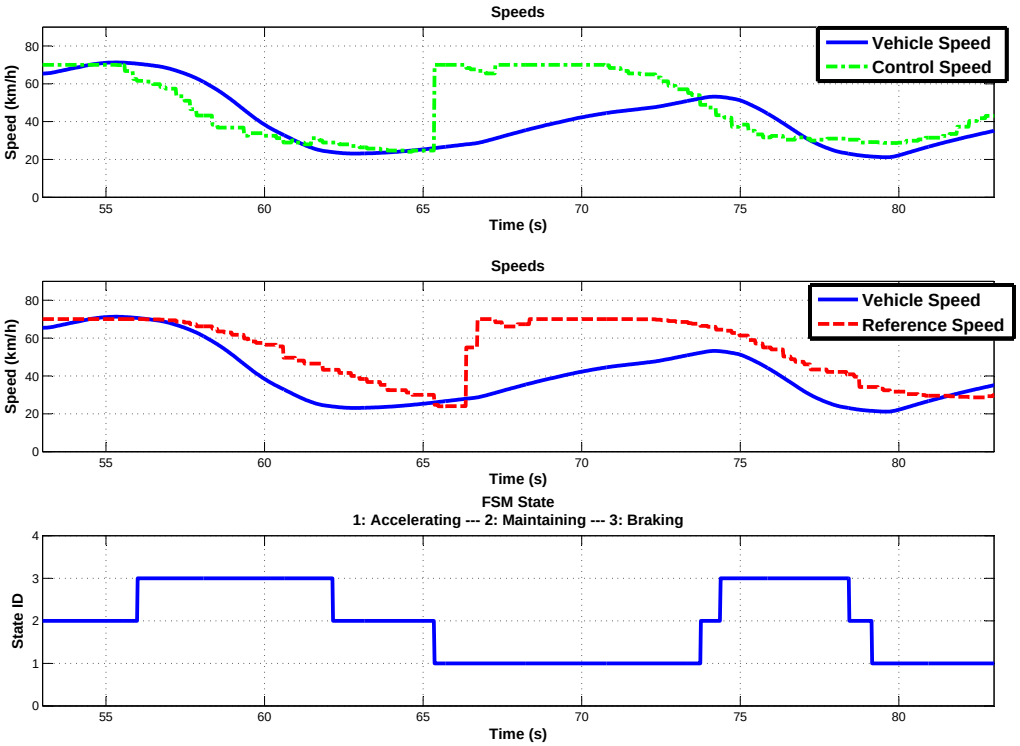



Figure 5.9: Speeds and FSM States Results

presented in Section.3.4.2.1. In addition, the different *Vehicle* positions ($VP$) are presented and, for each $VP$, different characters have been used to differentiate the states of the control system. *Accelerating*, *Maintaining* and *Braking* phases are, consequently, respectively marked by blue points, green squares and red diamonds. It is clear that the *Vehicle* behaves as foreseen: it is braking before both bends so that the *Vehicle* enters in them with an appropriate speed. Then it maintains its speed for a short period and finally re-accelerates.

This is confirmed by the elements presented in Fig.5.9. The top plot of this figure shows the *Vehicle* speed ($V$) and the control speed ($V_{ref}\,(X_{LAD}, Y_{LAD})$), in other words the reference speed determined with the prediction provided via the $LAD$ technique which is used by the Longitudinal Controller. The second plot helps to compare the speed of the *Vehicle* with the real speed reference ($V_{ref}\,(X_{VP}, Y_{VP})$), so to check if the *Vehicle* enters with the appropriate speed in the considered bends. Finally, the last plot presents the $FSM$ states. Note that the *Accelerating*, *Maintaining* and *Braking* states respectively correspond to 1, 2 and 3.

Lets now focus on the Fig.5.9's third plot: at the beginning, the control speed is close to the *Vehicle* speed, the corresponding state is then *Maintaining*. A few seconds later, as the control speed decreases due to the presence of the first bend, the difference between both speeds increases up to a value exceeding the speed tolerance area. Consequently, the system switches from the *Maintaining* state into the *Braking* state. The *Vehicle* speed then decreases until it becomes smaller than the control speed, so switching back into the *Maintaining* state. Further to this, the *Vehicle* reaches the bend, the control speed is then increasing as it looks forward so involves a change of state: *Maintaining* to *Accelerating*. This cycle then restarts for the second bend. Nevertheless, it is clear that the prediction provided by the $LAD$ technique helps to regulate the *Vehicle* speed safely. Indeed, the *Vehicle* speed is almost always lower than the reference

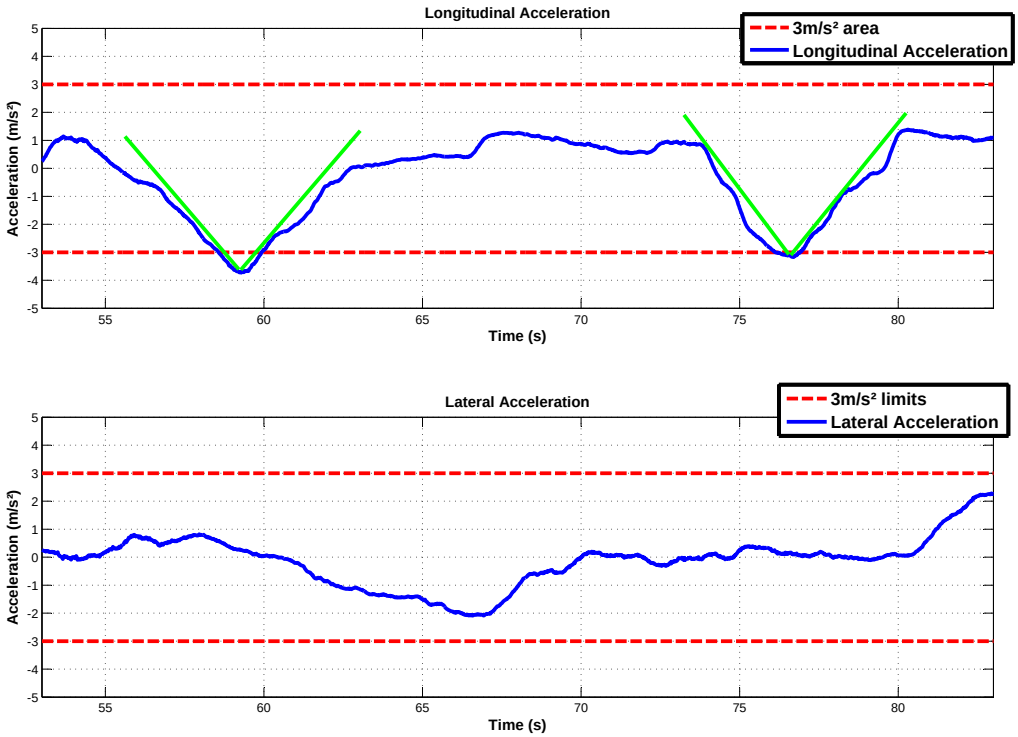



Figure 5.10: Measured Accelerations

speed, as presented in the Fig.5.9 second plot, so proving the safe behavior of the Longitudinal Controller. It can also be noted that the shape of the reference speed (and of the control speed) before bends is similar to the one presented in Fig.3.8 top plot. This proves the benefits of the limit speed profile modification which has been integrated to reproduce a smooth *Driver* behavior.

To validate the system hypotheses, the measurement of the longitudinal and lateral acceleration is necessary. Fig.5.10 presents the longitudinal acceleration (on the top plot) and the lateral acceleration (on the lower plot). The Braking states obviously involves the activation of the Active Braking System, which then implies an increase of the braking pressure gradually up to a maximum and then decreases it gradually to its initial value. The corresponding longitudinal acceleration follows the same behavior which is almost comparable to the theoretical deceleration triangle as shown by the green lines in Fig.5.10 top plot. At second 59 and 77, the Braking phase presents a maximal longitudinal acceleration of respectively $-3.2m/s^2$ and $-3.7m/s^2$ as no upper limit on $\gamma_L$ has been applied for safety reasons. Then, it is important to note that the lateral acceleration is respecting the limits for the considered road.

### 5.3.3 Conclusions

From this test, three main remarks can be done:

- The developed Longitudinal Controller, through the addition of several techniques such as the limit speed profile modification, the speed tolerance area and the *LAD* technique, provides a safe and comfortable speed regulation on open roads.
- The generation of the limit speed profile, which gives a set of potential reference for the

Longitudinal Controller, has been shown to be unadapted for direct application. Indeed, it is generated only based on geometrical information related to the road (the curvature) and a fixed maximal lateral acceleration value, so do not consider constraints related to the *Vehicle* kinematics and dynamics. These constraints have consequently been considered into the Longitudinal Controller synthesis step. This highlights the limitations of the unconstrained trajectory generation and the relevancy of the *Constraints Management* approach developed in this PhD.

- Nevertheless, the used mathematical model, $PCS$, already provides some advantages, e.g. its curvature continuity along the considered road which allows to define a variable speed profile. Furthermore it can be used for road of different composition: straight lines, bends, etc.

## 5.4 Constrained Trajectory Generation Results

This section is dedicated to the description of the results obtained using the constrained trajectory generation. As for the unconstrained trajectory generation, the presentation is divided into two parts: first simulation results are shown before the presentation of the results obtained after its application to the Lateral Control of a car-like *Vehicle*.

### 5.4.1 Simulations

#### 5.4.1.1 Tests Conditions

The simulations of the constrained trajectory generation have been done partly on the race track presented in Fig.5.1. For clarity reasons, the figures presented in the next sections correspond to a focus on relevant parts of the race track. These focuses, named *Area 1* and *Area 2*, respectively correspond to the succession of right and left bends and to the long right bend. Furthermore, only the validity area boundaries and the trajectories to be compared are represented (the road boundaries have been removed). Finally, the considered driving lane is the right lane: the validity area is computed in this part of the road.

#### 5.4.1.2 Constrained Trajectory Results

Fig.5.11 depicts the results obtained in *Area 1*. This figure presents, on the one hand, the results of the constrained trajectory generated WithOut Criterion Minimization ($WOCM$) and, on the other hand, a Spline which interpolates shape points located in the Center of the Lane ($CL$), corresponding to the unconstrained trajectory generated with linear parameter values. For clarity reasons, the figure has been rotated by 90°. Both trajectories are kept in the validity area and so, satisfy the geometric constraints linked to the road. Secondly, there is hardly any difference between them: they are both located near the middle of the lane. This is confirmed by the curvature comparison depicted in Fig.5.12. These figures tend to show that the $WOCM$ trajectory (which only grants the constraints) gives results similar to those of the $CL$ trajectory. However, the major difference lies in the approach: if the normal Spline interpolation requires the accurate location of the different shape points and so, defines a unique solution based on *a priori* knowledge, the constrained trajectory generation looks into all the possible trajectories allowed by the validity area and selects one. Consequently, the constrained trajectory is less sensitive to positioning errors and to *Digital Map Database* errors. There is also less *a priori* information required.

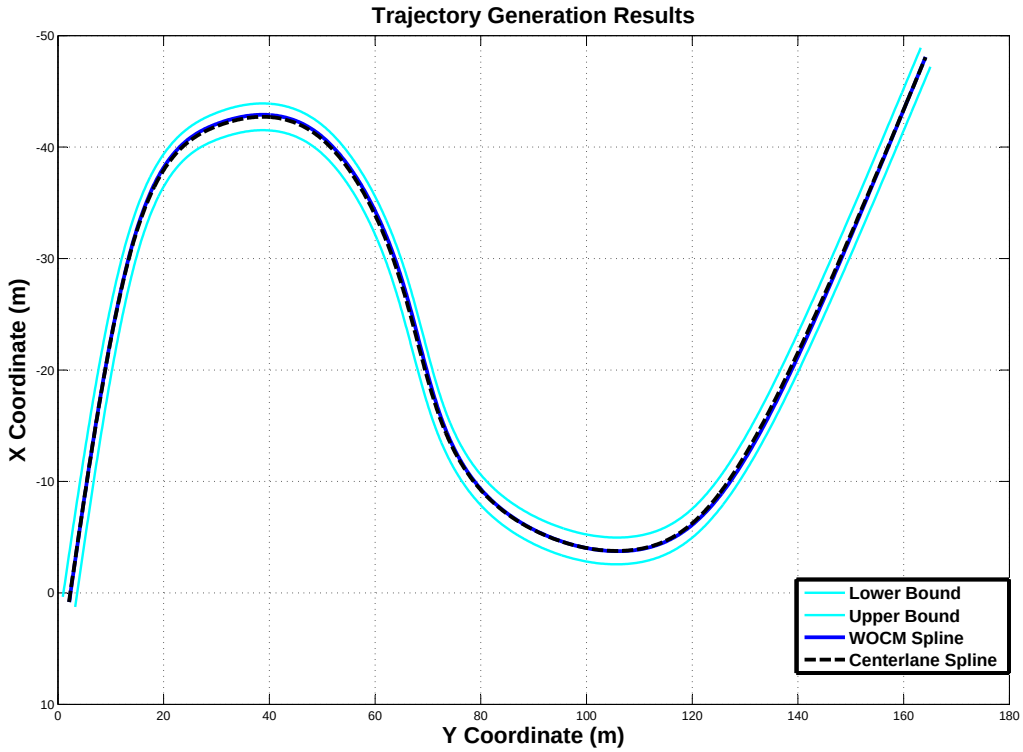



Figure 5.11: $CL$ vs $WOCM$ Trajectories in *Area 1*

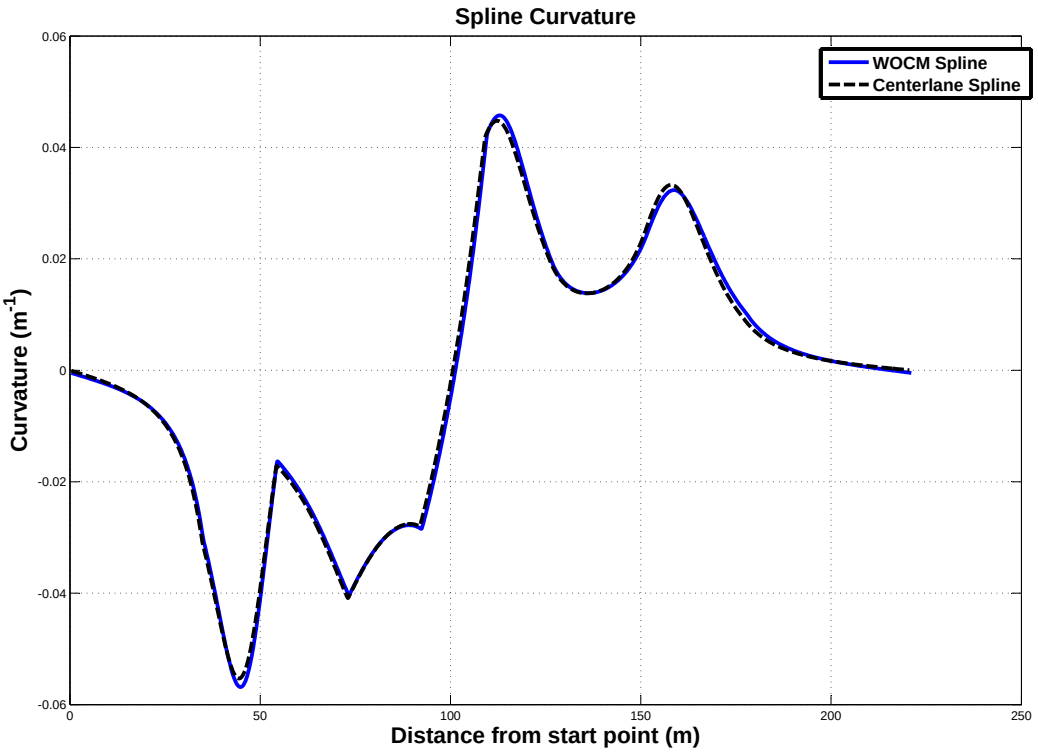



Figure 5.12: $CL$ vs $WOCM$ Curvatures in *Area 1*

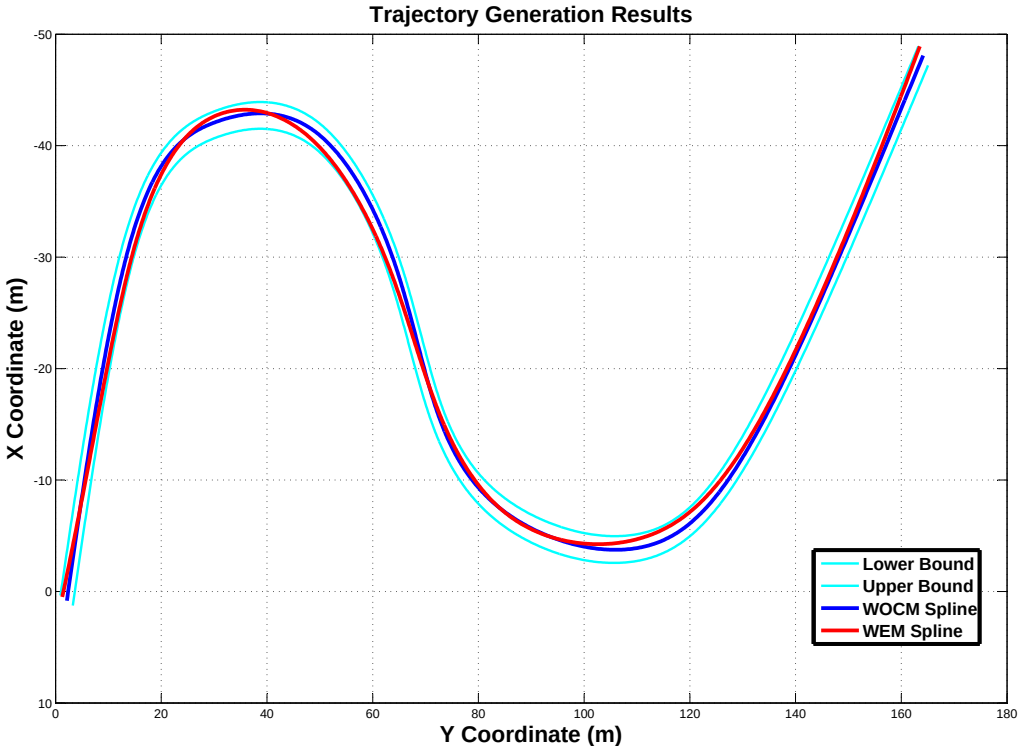



Figure 5.13: $WOCM$ vs $WEM$ Trajectories in *Area 1*

#### 5.4.1.3 Cost Criterion Minimization Results

Four cost criteria have been considered: no cost criterion ($WOCM$), the trajectory distance ($WDM$), the distance to a pre-defined reference ($WDRM$) and the trajectory energy ($WEM$). Remind that the $WDRM$ trajectory tries to minimize its distance with the unconstrained trajectory, so to the centerlane trajectory. The comparison between the different cost criteria is done in three steps: first the different trajectories are presented. For clarity purpose, their presentation are each time divided into two figures: one for the comparison of the $WOCM$ and $WEM$ trajectories and one for the $WDM$ and $WDRM$ trajectories. Then their respective curvature are depicted and, finally, a numerical comparison is provided through tables. These steps are processed for three road configurations: the two first are related to the race track used previously (so corresponds to *Area 1* and *Area 2*) and the last represents an example of road extracted from the *Digital Map Database* of $ADASRP$.

**Results in Area 1** Fig.5.13 and Fig.5.14 respectively present the results of the $WOCM$ versus $WEM$ trajectories and the $WDM$ versus $WDRM$ trajectories in area *Area 1*. These figures clearly show that all the trajectories are contained in the validity area and contrary to the previous figure (Fig.5.11), the differences between trajectories are here more visible. Indeed, the $WDRM$ trajectory (Fig.5.14), as it tries to match the unconstrained trajectory, is very close to the center of the validity area and so to the $WOCM$ trajectory. In addition, it is also clear that the $WDM$ trajectory is close to the interior parts of the bends as it represents the shortest path. Finally, the $WEM$ trajectory (Fig.5.13) uses the available lateral area with greater efficiency than the other trajectories: it is located alternately in the interior and in the exterior of the validity area.

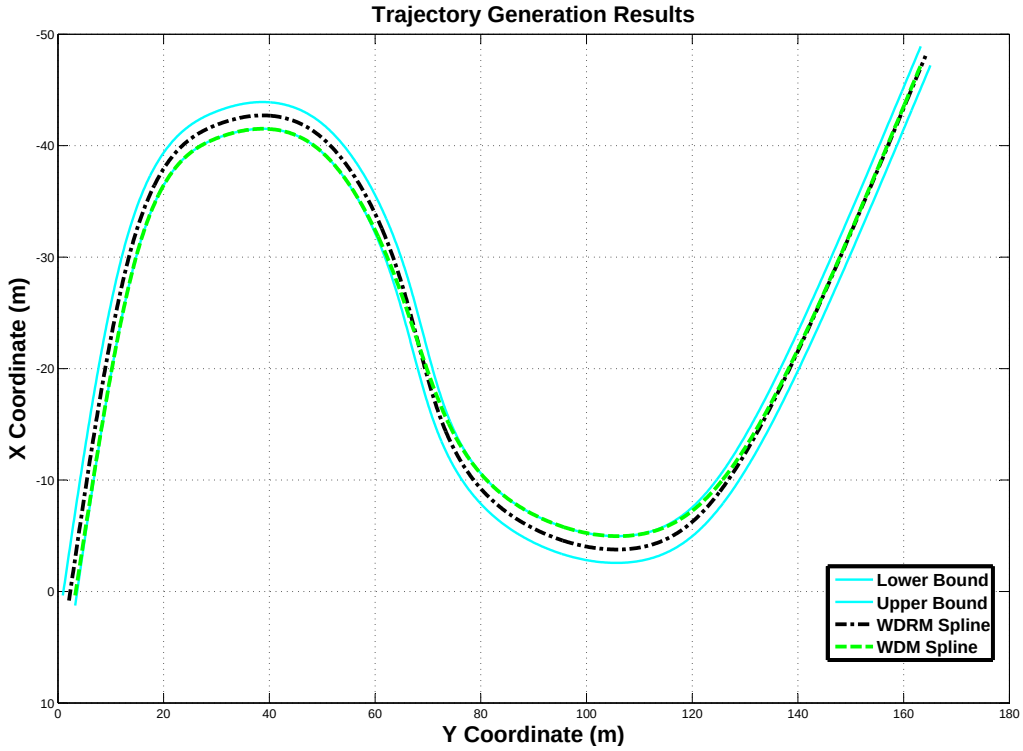



Figure 5.14: $WDM$ vs $WDRM$ Trajectories in *Area 1*

In Fig.5.15, the differences between the trajectories are confirmed. Indeed as suggested by the previous figures, the $WOCM$ trajectory curvature and the $WDRM$ trajectory curvature are almost overlapped, so proving the similarity of both trajectories. Then, as it could have been guessed, the curvature of the $WEM$ trajectory has lower amplitudes and is relatively smoother than the other trajectory curvatures especially in the first bend (between $25m$ and $100m$). Finally, it can be noted that the distance minimization does not seem to have a large impact on the curvature as the $WDM$ trajectory curvature as a similar shape compared to the $WOCM$ and the $WDRM$ trajectories.

Table.5.1 presents the different results obtained on *Area 1*. These results are related to two main elements:

1. *The comparison of the trajectory lengths and trajectory energies.* For this point, Table.5.1 presents values which validate the different remarks obtained from the study of the preceding figures. Indeed, the shortest trajectory seems to be the $WDM$ trajectory and the trajectory which seems to have the smallest energy is the $WEM$ one. Then, the energy reduction of the $WEM$ trajectory is quite significant as it goes up to 15% compared to the $WOCM$ trajectory.

2. *The post-checking of the constrains.* Indeed, as mentioned in Section 3.5.2.5, a post-checking of the constraints on the curvature and on the curvature derivative was carried out for each test. A threshold of $0.09m^{-1}$ was taken for the constraint linked to the minimum curvature of the car (3.13). It corresponds to an average turn radius of $11m$. The threshold of the constraint linked to the curvature derivative is of $0.038m^{-1}.s^{-1}$. It was obtained using (3.39) with a wheelbase $b$ of $2.5m$, a speed $v$ of $30km.h^{-1}$ (speed based on the

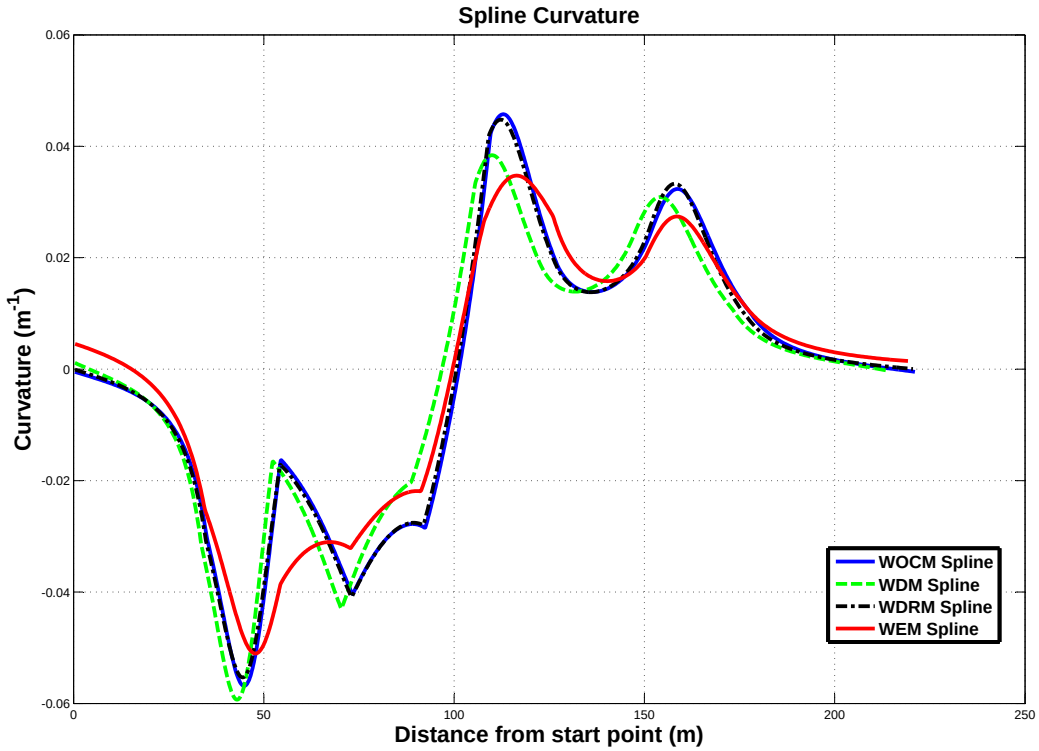



Figure 5.15: Trajectory Curvatures in *Area 1*

configuration of the bends), and a maximum steering angle speed of $15.7 rad.s^{-1}$ obtainable with an electrically driven steering wheel for lateral control (cf. [Pouly, 2009]). Finally, the threshold of the constraint linked to the car acceleration ($\gamma_{T_{max}}$) was fixed to the common lateral acceleration value used for comfortable driving: $3m.s^{-2}$. These three constraints are respectively marked $\kappa_{max}$, $\dot{\kappa}_{max}$ and $\gamma_{max}$ in the tables.

This table shows that the maximal curvature constraint allowed by the car geometry is respected by all trajectories. This is also the case for the maximal steering speed. On the other hand, the effects of the energy minimization can also be noticed in this table. Indeed, the maximum curvature value of the $WEM$ trajectory seems to be lower than the others.

However, the maximal allowed acceleration constraint, is not provided by any trajectory for a constant speed of $30km.h^{-1}$. Nevertheless, the best results are obtained with the $WEM$ trajectory. Fig.5.16 shows that this constraint is globally satisfied except the transient overshoots at the entrance of the first bend around $45m$ and of the second bend $75m$. This shows that a constant $30km.h^{-1}$ speed is not suited to track the present trajectories considering this road configuration. To cope with this problem, a speed reduction could be envisaged. A reduction of $3km.h^{-1}$ for *Area 1* would help to satisfy the constraint on the whole $WEM$ trajectory, contrary to the other trajectories which require a $5km.h^{-1}$ reduction. Another solution could be the consideration of a variable speed. Indeed, from the application of the unconstrained trajectory generation on a Longitudinal Control, it has been shown that a variable speed profile based on the information provided by a Spline trajectory can be used to manage the *Vehicle* speed dynamically.

Table 5.1: Numerical Results for Area1

| | $TrajectoryLength(m)$ | $Energy$ | $k_{max}(m^{-1})$ | $\dot{k}_{max}(m^{-1}.s^{-1})$ | $\gamma_{max}(m.s^{-2})$ |
|---|---|---|---|---|---|
| $CL$ | 221 | 0.13 | 0.06 | 0.001 | 3.84 |
| $WOCM$ | 222 | 0.13 | 0.06 | 0.001 | 3.95 |
| $WDM$ | **215** | 0.12 | 0.06 | 0.001 | 4.12 |
| $WDRM$ | 221 | 0.13 | 0.06 | 0.001 | 3.84 |
| $WEM$ | 220 | **0.11** | 0.05 | 0.001 | 3.54 |
| $Threshold$ | | | **0.09** | **0.038** | **3.00** |

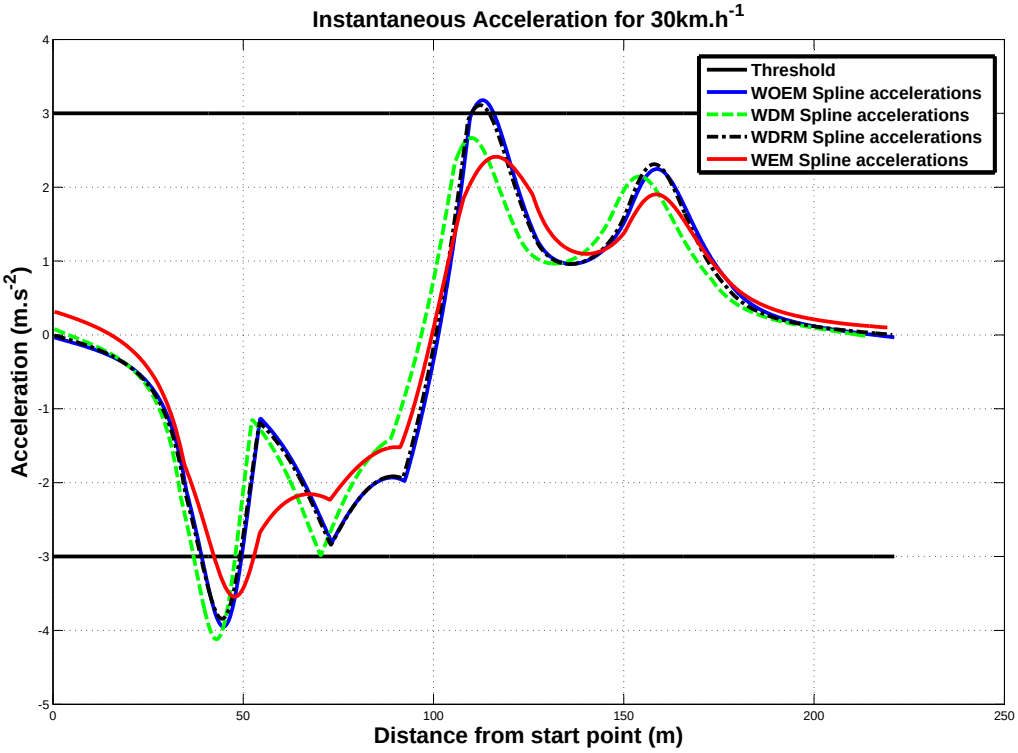



Figure 5.16: Trajectory Instantaneous Acceleration in *Area 1*

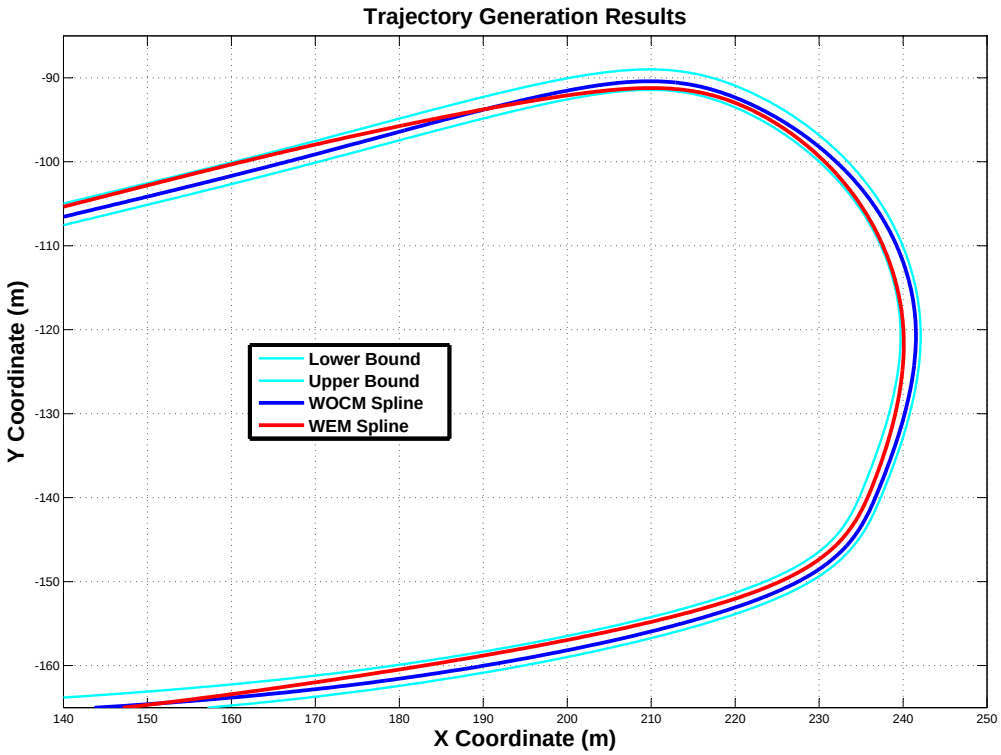



Figure 5.17: $WOCM$ vs $WEM$ Trajectories in *Area 2*

**Results on Area 2** Fig.5.17 and Fig.5.18 respectively present the results of the $WOCM$ versus $WEM$ trajectories and the $WDM$ versus $WDRM$ trajectories in area *Area 2*. The behavior of the different trajectories are similar to those generated on *Area 1*: all the trajectories are contained in the validity area, the $WDRM$ trajectory is very close to the center of the validity area; the $WDM$ trajectory is close to the interior parts of the bend as it looks for a minimization of the total length; the $WEM$ uses the available lateral area with greater efficiency than the other trajectories. These behaviors seem to be confirmed by the trajectory curvatures presented in Fig.5.19.However, note that the reduction of the trajectory length provided by the $WDM$ trajectory is less visible, thus less important than the one provided in *Area 1*. Furthermore, the smoothing brought by the energy minimization also seems to be slight here.

Table.5.2 presents the numerical results obtained on *Area 2* similarly to Table.5.1. These results confirm the remarks obtained through the analysis of the trajectories and their curvatures: the $WDM$ and the $WEM$ trajectories benefits are almost not visible. This proves that the effectiveness of the constrained trajectories regarding to their respective criterion depends on the initial shape of the road. In fact, the test results described in this section were obtained in circular bends. The $CL$ and $WOCM$ trajectories which are located near the center of the validity area (so near the center of the lane) are also already close to the minimum energy curve. The potential reduction of the $WEM$ trajectory is consequently limited. Finally remember that the considered solution set, which defines the inequalities of the different trajectories, is limited to subset $A$. Better results could be expected, considering $A \cup B$ (cf. Section.3.5.2.2).

For the post-checking of the kinematic and dynamic constraints, the results are similar between the trajectories generated on *Area 1* and on *Area 2*: the first constraint which corresponds to the maximal curvature allowed by the car geometry, is respected by all trajectories. This is

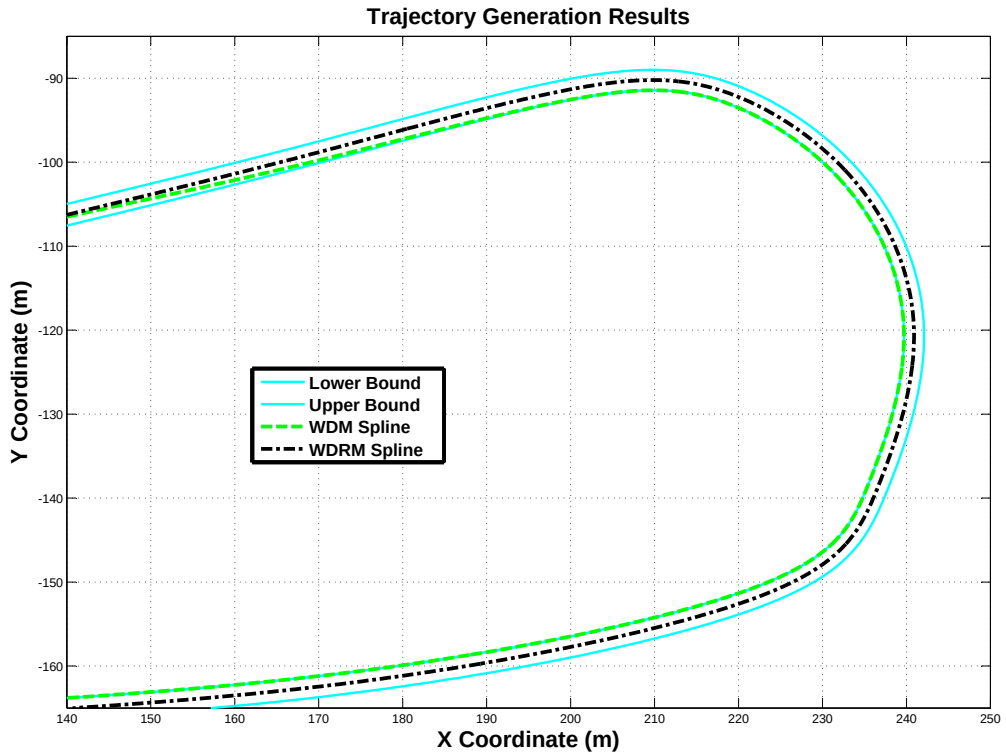



Figure 5.18: $WDM$ vs $WDRM$ Trajectories in *Area 2*

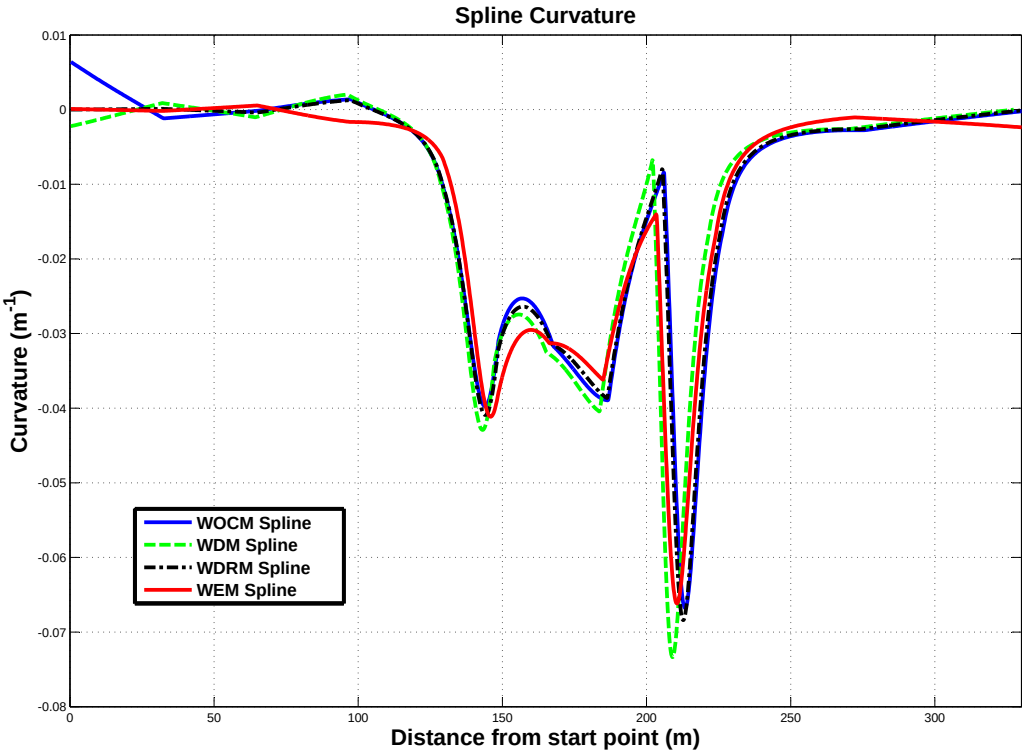



Figure 5.19: Trajectory Curvatures in *Area 2*

Table 5.2: Numerical Results for Area2

| | $TrajectoryLength(m)$ | $Energy$ | $k_{max}(m^{-1})$ | $\dot{k}_{max}(m^{-1}.s^{-1})$ | $\gamma_{max}(m.s^{-2})$ |
|---|---|---|---|---|---|
| $CL$ | 333 | 0.11 | 0.07 | 0.002 | 4.75 |
| $WOCM$ | 334 | 0.11 | 0.07 | 0.002 | 4.70 |
| $WDM$ | **328** | 0.11 | 0.07 | 0.002 | 5.09 |
| $WDRM$ | 333 | 0.11 | 0.07 | 0.002 | 4.75 |
| $WEM$ | 330 | 0.11 | 0.07 | 0.002 | 4.59 |
| $Threshold$ | | | **0.09** | **0.038** | **3.00** |

Table 5.3: Numerical Results for a Digital Map Bend

| | $TrajectoryLength(m)$ | $Energy$ | $k_{max}(m^{-1})$ | $\dot{k}_{max}(m^{-1}.s^{-1})$ | $\gamma_{max}(m.s^{-2})$ |
|---|---|---|---|---|---|
| $CL$ | 126 | 0.19 | 0.11 | 0.003 | 7.51 |
| $WOCM$ | 126 | 0.19 | 0.11 | 0.003 | 7.62 |
| $WDM$ | **124** | 0.19 | 0.11 | 0.003 | 7.62 |
| $WDRM$ | 126 | 0.19 | 0.11 | 0.003 | 7.51 |
| $WEM$ | 126 | **0.15** | 0.07 | 0.002 | 4.89 |
| $Threshold$ | | | **0.09** | **0.038** | **3.00** |

also true for the maximal steering speed. Nevertheless, it can be noted that the energy minimization has here no impact. This in confirmed by the third constraint, i.e. the maximum acceleration value, which depends on the curvature maximum. In addition, the third constraint is not directly granted by any of the generated trajectory for a constant speed of $30km.h^{-1}$.

**Using Classical Digital Map Data** In the previous sections, the different trajectory generation techniques were studied on a test track composed of regular bends separated by straight lines. This section is devoted to the comparison of trajectories on open roads digitalized with the inaccuracies described in Section.2.3.3. As for previous figures and for clarity reasons, only the boundary curves and the relevant trajectories are presented.

Results of the constrained trajectory generations on a non regular bend are presented in Fig.5.20 and Fig.5.21. Similarly to the results presented in the previous sections, all the trajectories are kept within the validity area. In addition, the behavior of the different trajectories are still similar to the previous tests as the $WDM$ trajectory almost overlaps the center part of the bend and as the $WEM$ trajectory uses the lateral space more efficiently than the other trajectories. However the major difference lies in the curvatures presented in Fig.5.22. Indeed, it shows an important smoothing of the curvature for the $WEM$ trajectory, so confirming its benefits compared with conventional approaches ($CL$, $WOCM$). This is confirmed by Table.5.3 in which, the effect of the energy minimization is clearly shown: respective average reduction of 21% 36% 33% 36% for the energy, the curvature maximum, the curvature derivative maximum and the maximum acceleration compared to the $CL$ trajectory[27]. These figures and tables confirms that the benefits of the constrained trajectories are not only valid for the specific race track used in the previous section, but for a large panel of road configurations.

[27]The checking of the constraints has been done using the same conditions: same speed of $30km.h^{-1}$ and same constraint thresholds.

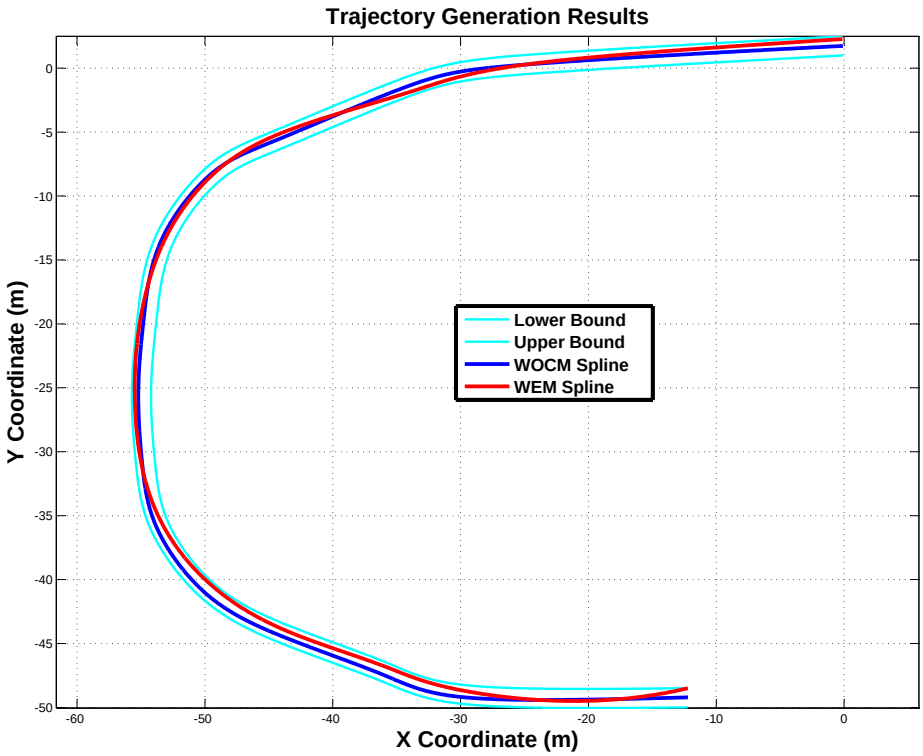



Figure 5.20: $WOCM$ vs $WEM$ Trajectories in a Non-regular Bend

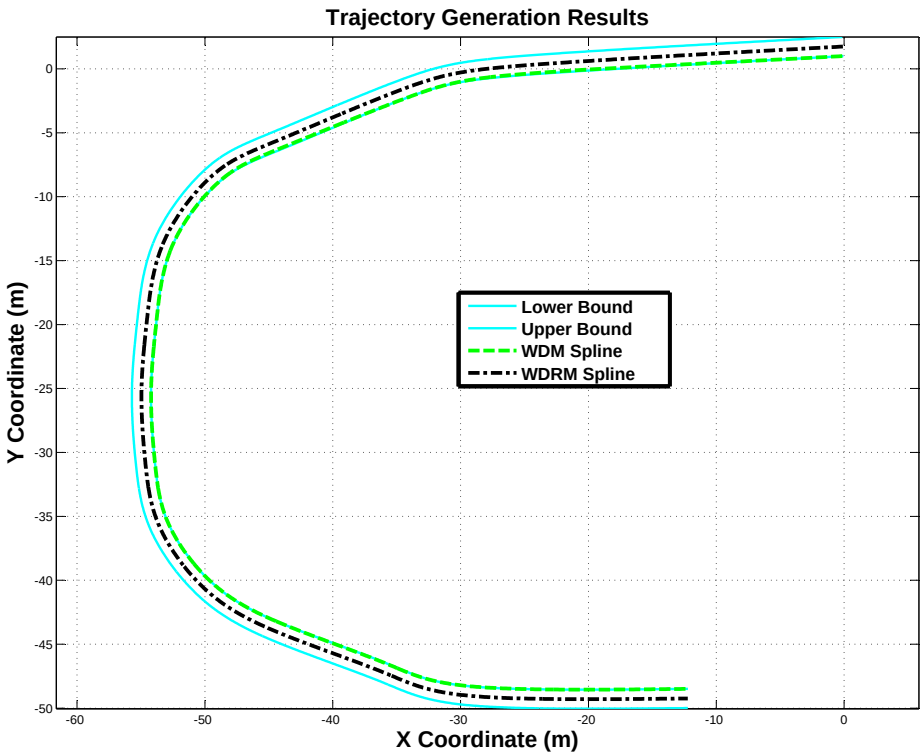



Figure 5.21: $WOCM$ vs $WEM$ Trajectories in a Non-regular Bend

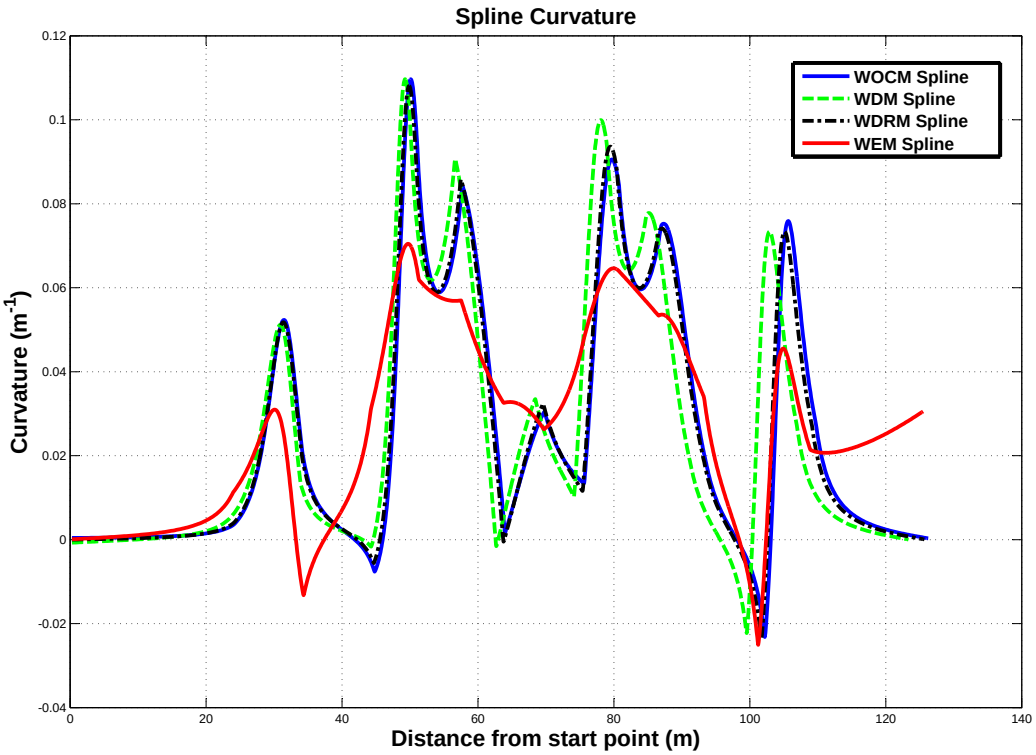



Figure 5.22: Trajectory Curvatures in a Non-regular Bend

### 5.4.2 Lateral Control Results

The following paragraph presents the results obtained with the Lateral Control using the aforementioned trajectories. The Lateral Controller is based on the Model Predictive Control ($MPC$) presented in [Pouly, 2009] (cf. Section.3.5). Details about this controller synthesis are available in Appendix.C.

#### 5.4.2.1 Evaluation Conditions

A comparison of the electric steering power required to track each trajectory is provided here. To determine the estimated power, a complete $3D$ *Vehicle* model has been used. The $LPV/LTI$ model used for the control is a linearization of this $3D$ model. This simplification was performed to match the real-time requirements for the $MPC$ control. Then, considering no losses, the expression of the electric power required by the Steering Motor (cf.A.2) to track the different trajectories is defined by:

$$P_{Electric} = U \cdot I = U \cdot \frac{C}{k_m} \tag{5.5}$$

with $U = 24V$ the steering motor voltage, $k_m = 4.73$ the coefficient which links the steering motor torque $C$ to the current $I$ given by the steering motor designer.

The Lateral Controller tracking has been tested on the left bend presented in Fig.5.20 using two strategies:

- The first test has been obtained considering a constant speed of $30km.h^{-1}$ along the trajectories.

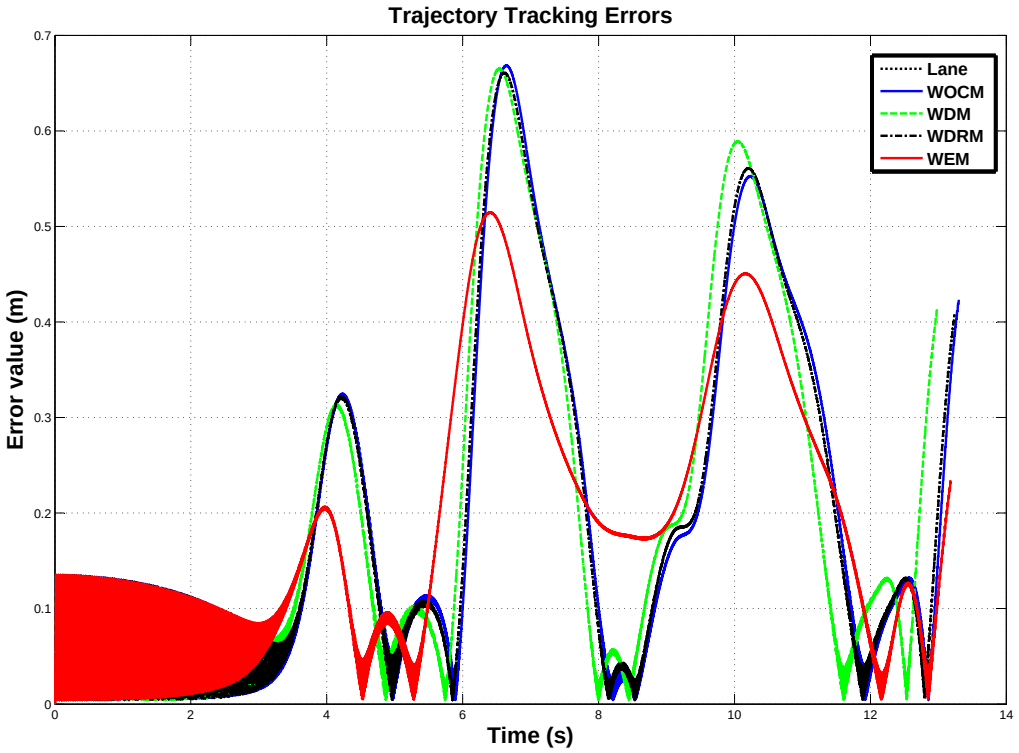



Figure 5.23: Tracking Error with a Fixed Speed ($30km.h^{-1}$)

- The second test has been obtained considering a variable speed along the trajectories. The latter has been obtained using the reference speed model presented in (3.7).

#### 5.4.2.2 Results

The trajectories which have been used for the Lateral Controller are those presented in Fig.5.20 and in Fig.5.21: the trajectory WithOut Criterion Minimization ($WOCM$), With Distance Minimization ($WDM$), With Distance to Reference Minimization ($WDRM$), and With Energy Minimization ($WEM$). The unconstrained trajectory ($CL$) has also been tested here for comparison purposes.

**Constant Speed Results** Fig.5.23 and Fig.5.24 respectively presents the tracking error, in other words the distance between the *Vehicle* position and the reference trajectory, and the control signals generated by the $MPC$ Controller for a constant speed over the considered bend.

The first remark which can be done is that the tracking of all the trajectories is quite efficient. Indeed, the overall maximum tracking error, obtained with the $WOCM$ trajectory is of $0.7m$ with a mean value of $0.2m$ (cf. Table.5.4 values for the *Left Bend*), and this even if the precision of the Lateral Controller has been penalized - $Q = 1$ and $R = 2$; better precision could be expected by increasing the weight of $Q$, especially at the beginning of the test which presents tracking error oscillations around $0.1m$. These oscillations are due to the small weight of the precision and to the fact that the *Vehicle* tries to follow the straight line situated at the beginning of the test road. Then, it is clear that the $CL$, $WOCM$, $WDM$ and $WDRM$ trajectories present close or equivalent tracking results contrary to the $WEM$. Indeed, the $WEM$ trajectory seems to be more efficient: a maximum tracking error of $0.5m$ with a mean value of $0.2m$, and this

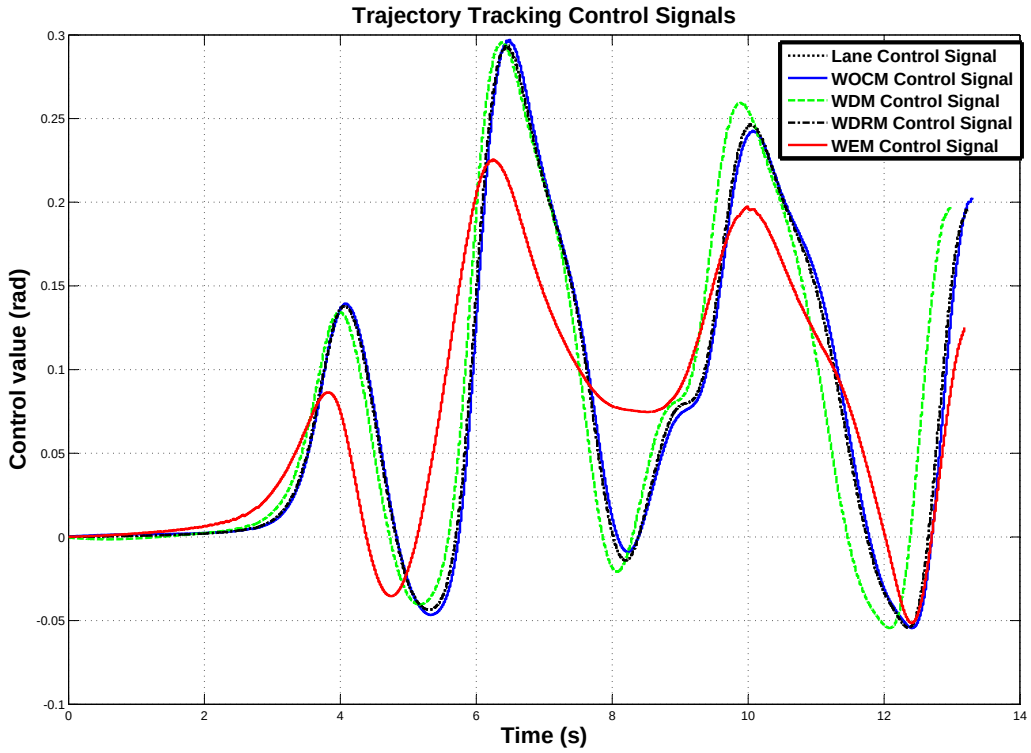



Figure 5.24: Control Signals Generated by the Lateral Controller for a Fixed Speed ($30km.h^{-1}$)

with lower control values (so lower wheel angles) as presented in Fig.5.24. Here again, the other trajectories have close results. It can be interesting to note that the control signals and the tracking errors have a shape which is very close to the trajectory curvature shapes presented in Fig.5.22. This proves that the curvature and its continuity are essential elements for the considered Lateral Controller. This may explain the better efficiency of the $WEM$ trajectory. Indeed, as the $WEM$ trajectory already integrates some of the *Vehicle* kinematics and dynamics limitations (cf. Section.3.5.1) through the minimization of the trajectory energy, this path may consequently not contain portions in which the *Vehicle* reaches or overcomes its limitations, in other words, portion in which the *Vehicle* is not able to follow the trajectory. That is why the $WEM$ trajectory presents a smoother control signal and involves less tracking errors. The last remark on Fig.5.24 can be done on the $WDM$ trajectory. Indeed, considering iso-conditions, the $WDM$ tracking ends a short time before the other trackings.

Fig.5.25 presents the electric power consumption implied by each trajectory tracking along the considered bend. These curves have shapes which are equivalent to those presented in the two previous figures. The observations made on the previous figures are consequently still valid: the $WEM$ trajectory tracking presents a smoother and a lower amplitude compared to the others.

This seems to be confirmed by Table.5.4 which presents numerical tracking results over several roads - the first one corresponding to the presented left bend. Indeed, the $WEM$ trajectory tracking errors are similar or lower than the others trackings. This tends to confirm the good properties of the $WEM$ trajectory, and this for all the bends. In addition, even if they are not shown here, tests done upon the sharp right bend have revealed that the $WEM$ trajectory, contrary to the others, avoids the generation of a control reaching the wheel angle limits of $\pm 26°$. This tends to show that the consideration of the constraints related to the *Vehicle* and

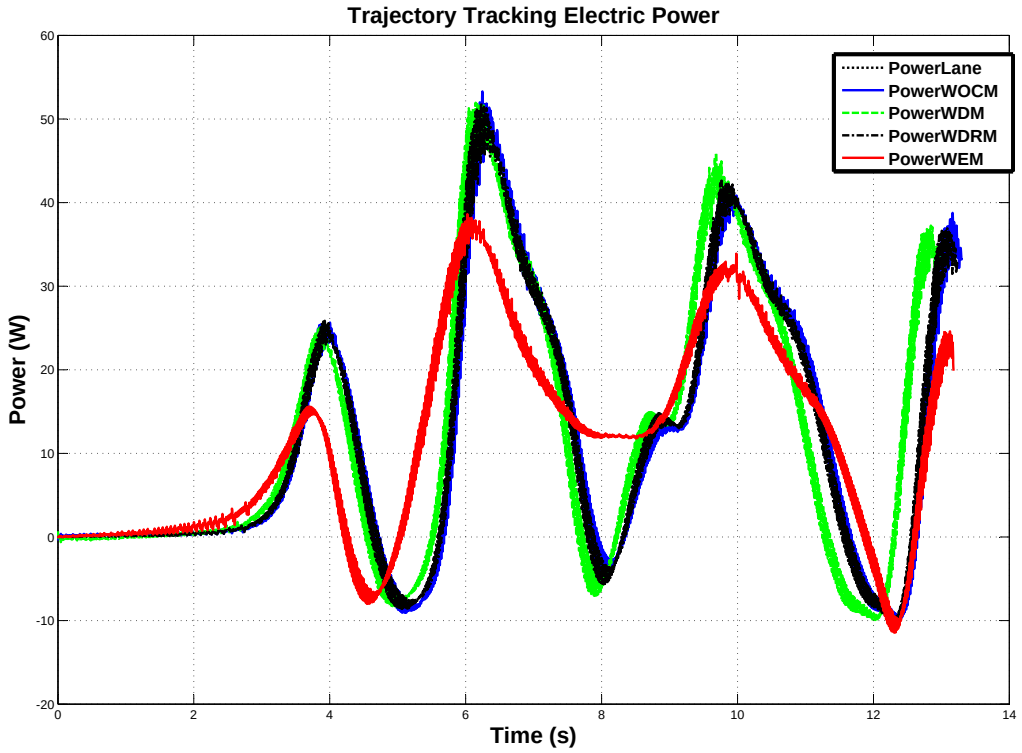



Figure 5.25: Instantaneous Power Consumption for a Fixed Speed ($30km.h^{-1}$)

Table 5.4: Lateral Control Results for a Fixed Speed ($30km.h^{-1}$)

| | | CL | WOCM | WDM | WDRM | WEM |
|---|---|---|---|---|---|---|
| Left Bend | MaxError (m) | 0.7 | 0.7 | 0.7 | 0.7 | 0.5 |
| | MeanError (m) | 0.2 | 0.2 | 0.2 | 0.2 | 0.2 |
| Sharp Right Bend | MaxError (m) | 1.5 | 1.5 | 1.6 | 1.5 | 0.9 |
| | MeanError (m) | 0.3 | 0.3 | 0.3 | 0.3 | 0.2 |
| Smooth Right Bend | MaxError (m) | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 |
| | MeanError (m) | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 |

its *Environment* directly in the generation of the *Reference* helps to generate trajectories which are more suited to the Lateral control approach.

**Variable Speed Results** In Section.5.4.1.3, a post checking of the constraints was proposed. The consideration of a constant speed to track these trajectories is inappropriate considering the lateral acceleration limit. A variable speed strategy, defined w.r.t. the trajectory to be followed, is consequently proposed. This variable speed is defined regarding the reference speed model presented in (3.7) saturating at $30km.h^{-1}$. The different speed profiles, which are presented in Fig.5.26, show here again, that the trajectories have close profiles except the $WEM$ trajectory. Indeed, since the speed is computed regarding the curvature, the $WEM$ speed profile presents higher values. Trajectory tracking results presented in this section have been obtained by considering that the *Vehicle* exactly matches the speed profiles presented in Fig.5.26.

Fig.5.27 and in Fig.5.28 present the tracking errors and the Lateral Controller signals obtained with variable speeds. Considering the test conditions, the fact that the $WEM$ tracking ends

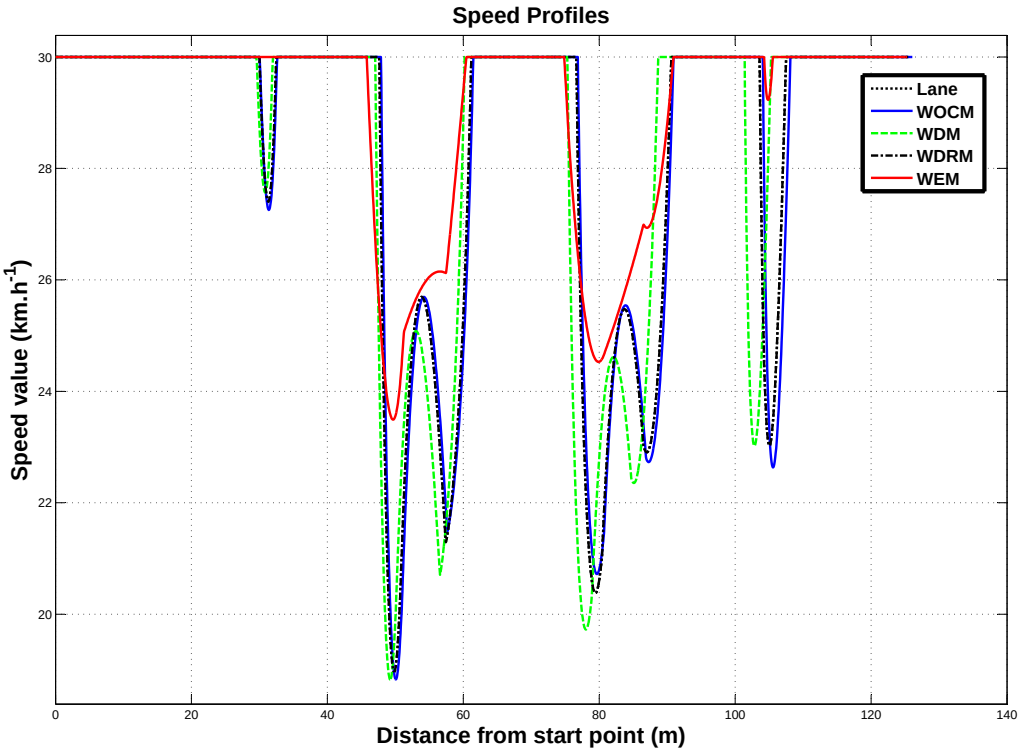



Figure 5.26: Speed Profiles

Table 5.5: Lateral Control Results for a Variable Speed

| | | CL | WOCM | WDM | WDRM | WEM |
|---|---|---|---|---|---|---|
| Left Bend | MaxError (m) | 0.8 | 0.8 | 0.8 | 0.8 | 0.6 |
| | MeanError (m) | 0.2 | 0.2 | 0.3 | 0.2 | 0.2 |
| Sharp Right Bend | MaxError (m) | 2.4 | 2.4 | 2.5 | 2.4 | 1.5 |
| | MeanError (m) | 0.4 | 0.4 | 0.5 | 0.4 | 0.3 |
| Smooth Right Bend | MaxError (m) | 0.6 | 0.6 | 0.6 | 0.6 | 0.6 |
| | MeanError (m) | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 |

slightly before the others is coherent. Indeed, the $WEM$ speed profile allows the *Vehicle* to go faster which consequently involves the *Vehicle* to reach the end of the bend earlier. Then, the $WEM$ trajectory presents the overall best efficiency in terms of tracking errors and in control signal amplitude, especially for the sharp right bend in which the tracking errors are reduced from $2.5m$ to $1.5m$ for the maximum value and from $0.5m$ to $0.3m$ for the mean value (see Table.5.29). In addition, as for the fixed speed test, Fig.5.29 shows that the $WEM$ trajectory presents lower power consumption amplitudes than the other trajectories.

The variable speed influence is confirmed by comparing Fig.5.25 and Fig.5.29. It can be noted that the level of instantaneous power consumption has been globally reduced. This may originate from the consideration of the lower speeds in bends. However, this variable speed tracking conditions involves an increase of the tracking errors as shown in Table.5.5. This tends to prove that the consideration of variable speed, if it helps to reduce the power consumption, should imply the modification of the $MPC$ controller $Q$ and $R$ parameters.

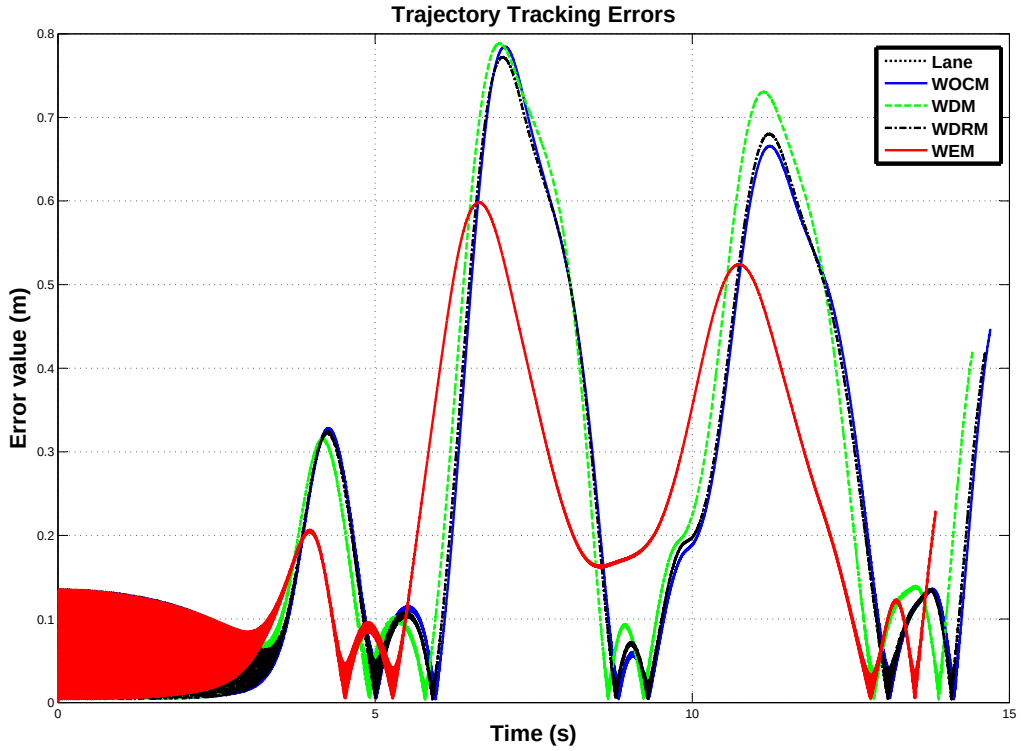



Figure 5.27: Tracking Error with a Variable Speed

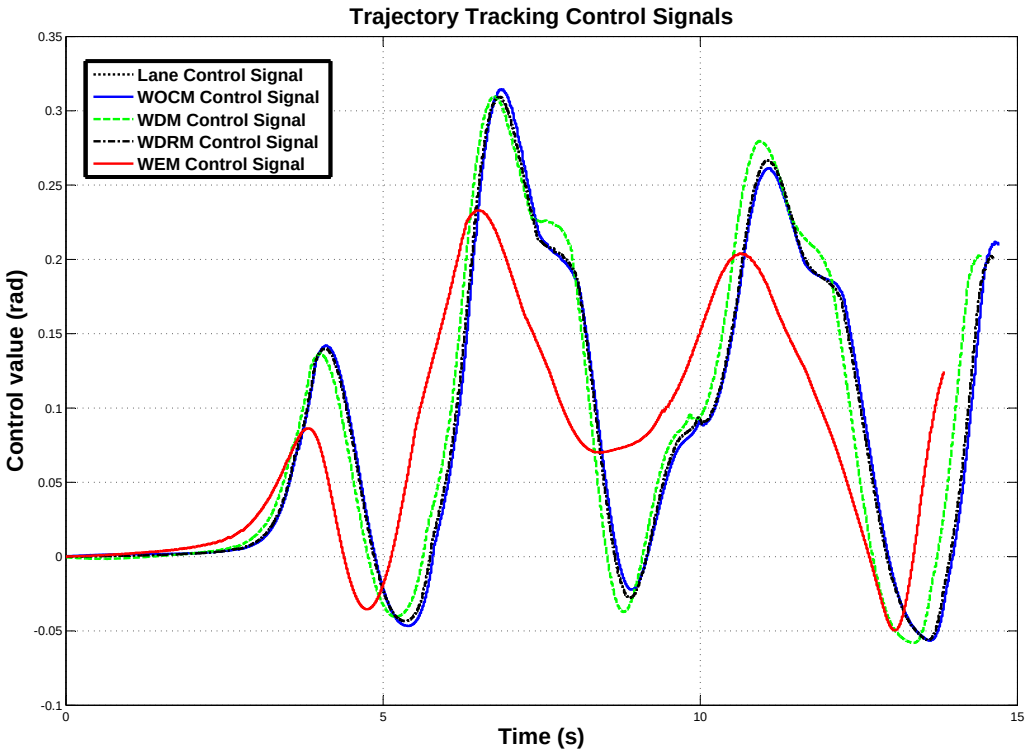



Figure 5.28: Control Generated by the Lateral Controller for a Variable Speed

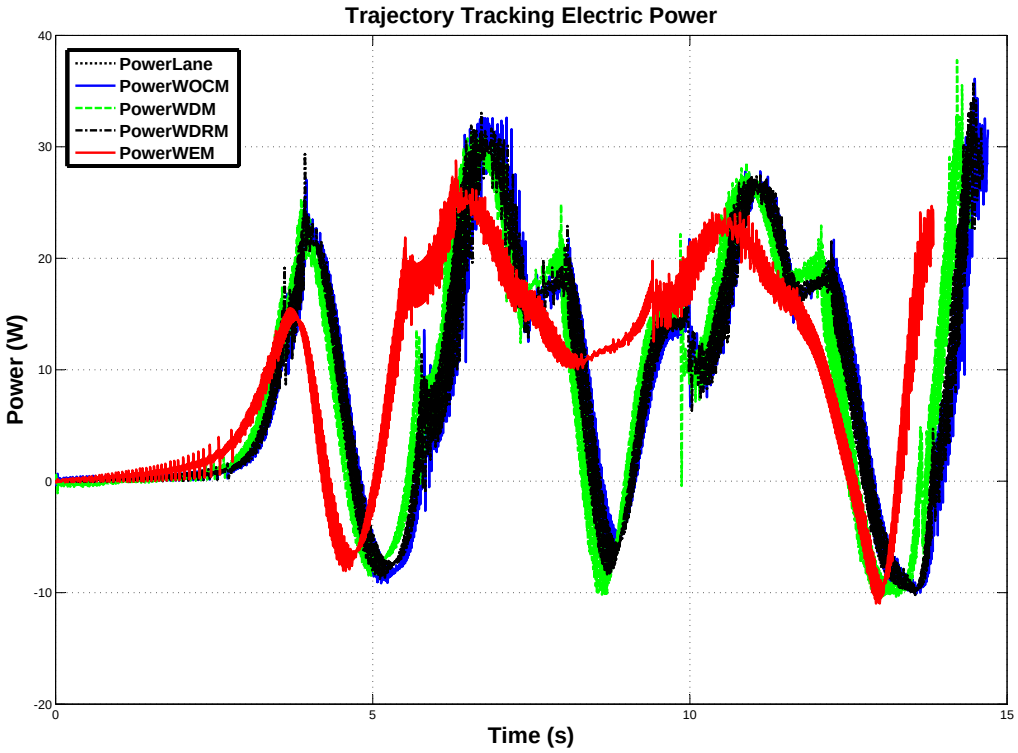



Figure 5.29: Instantaneous Power Consumption for a Variable Speed

### 5.4.3 Conclusion

The main information to be retained is that the consideration of the constraints related to the *Vehicle* directly in the generation of the *Reference*, in other terms the generation of constrained trajectories, helps to provide better control results. Indeed, the previous sections which have presented results obtained considering different configurations, have proved that the $WEM$ tracking generally involves less tracking errors, control values of lower amplitudes and finally consumes less energy. This reveals the relevancy of the present approach which consists in the distribution of the constraints over the *Reference* and over the *Controller* synthesis step.

## 5.5 Speed Limit Determination Results

This section is dedicated to the description of the results obtained using Evidence-based Data Fusion for Speed Limit Assistance. To clearly present the benefits of the proposed solution, this section is divided into four subsections. The two firsts are dedicated to the presentation of the benefits obtained with the multi-criterion fusion and the multi-sensor fusion in simulation. The third subsection depicts real-time experiments carried out with the test car presented in Appendix.A. The last section concludes the presentation of the results.

### 5.5.1 Discernment Frame Definition

The discernment frame used for Speed Limit Determination contains all the speeds which can be considered by the navigation and the vision (camera + $SLSR$) systems. It is obviously related to speeds defined by legal driving rules such as:

$$\Theta = \{5, 10, 20, 30, 45, 50, 60, 70, 80, 90, 100, 110, 120, 130, \textit{unlimited}\} \tag{5.6}$$

Very low speeds ($5km.h^{-1}$ to $30km.h^{-1}$) refer to situations which can only occur in small and specific in-city locations (car parks, school neighborhoods, etc.) while *unlimited* refers to roads which do not have any speed limitation. This mainly refer to German highways.

The vision modeling and mass estimation is directly processed over one of this speed. On the opposite, the navigation speed is obtained after the multi-criterion fusion. The latter first looks at the speed extracted from the *Digital Map Database* and then defines the *bba* over a set of focal speeds. This focal speed set is determined regarding the *Digital Map Database* speed such as presented in Appendix.D (cf. Table.D.1). This table transcribes the results of empirical tests. In other words, it refers to possible solutions when a navigation incoherency is detected.

### 5.5.2 Belief Masses Identification

As depicted in Chapter.4, the $SLA$ is composed of two fusion levels which are: the multi-criterion and multi-sensor fusion. These fusion steps obviously require the identification of the belief masses of the vision and the navigation information. The model considered for the vision and navigation *bba* is based on the one initiated by *Rombaut/Gruyer* (cf. Chapter.4).

The vision information is determined regarding the data provided by the $SLSR$ which returns the most probable detected speed and its probability. As mentioned previously, the probability can be considered as the vision confidence variable $C_{v_{vis}}$, which is decreased by a forgiveness factor $FF$. Then, by considering that the $SLSR$ is subject to a minimum ignorance value of 0.1, the vision information is determined by the linear expressions given in (5.7) regarding its confidence variable $C_{v_{vis}}$.

$$\begin{aligned}
m_v(H_v) &= \begin{cases} 0 & (C_{v_{vis}} - FF) \in [0, \tau] \\ \left(\frac{0.9}{1-\tau}\right)(C_{v_{vis}} - FF) - \frac{0.9\tau}{1-\tau} & (C_{v_{vis}} - FF) \in [\tau, 1] \end{cases} \\
m_v(H_v^c) &= \begin{cases} -\frac{0.9}{\tau}(C_{v_{vis}} - FF) + 0.9 & (C_{v_{vis}} - FF) \in [0, \tau] \\ 0 & (C_{v_{vis}} - FF) \in [\tau, 1] \end{cases} \\
m_v(\Theta) &= \begin{cases} \frac{0.9}{\tau}(C_{v_{vis}} - FF) + 0.1 & (C_{v_{vis}} - FF) \in [0, \tau] \\ -\left(\frac{0.9}{1-\tau}\right)(C_{v_{vis}} - FF) + \frac{1-(0.1\tau)}{1-\tau} & (C_{v_{vis}} - FF) \in [\tau, 1] \end{cases}
\end{aligned} \tag{5.7}$$

with $m_v(H_v)$, $m_v(H_v^c)$ and $m_v(\Theta)$ respectively corresponding to the belief mass on the speed, on the complementary speed and on the ignorance. $\tau$ is here equal to 0.5.

The navigation belief mass identification is based on the same model but the behavior is different. Indeed, the navigation *bba* is processed through the multi-criterion fusion (cf. Section.4.5.1). Consequently, for each criterion, a *bba* is defined by (5.8) regarding the confidence variable $C_{v_{nav}}$.

$$\begin{array}{rcl} m_{j,n}(H_n) & = & \begin{cases} 0 & C_{v_{nav}} \in [0,\tau] \\ \left(\frac{Crit_v}{1-\tau}\right) C_{v_{nav}} - \frac{Crit_v \tau}{1-\tau} & C_{v_{nav}} \in [\tau,1] \end{cases} \\ m_{j,n}(H_n^c) & = & \begin{cases} -\frac{Crit_v}{\tau} C_{v_{nav}} + Crit_v & C_{v_{nav}} \in [0,\tau] \\ 0 & C_{v_{nav}} \in [\tau,1] \end{cases} \\ m_{j,n}(\Theta) & = & \begin{cases} \frac{Crit_v}{\tau} C_{v_{nav}} + (1 - Crit_v) & C_{v_{nav}} \in [0,\tau] \\ -\left(\frac{Crit_v}{1-\tau}\right) C_{v_{nav}} + \frac{1-(1-Crit_v)\tau}{1-\tau} & C_{v_{nav}} \in [\tau,1] \end{cases} \end{array} \tag{5.8}$$

with $m_{j,n}(H_n)$, $m_{j,n}(H_n^c)$ and $m_{j,n}(\Theta)$ respectively corresponding to the belief mass on the speed $n$, on the complementary speed and on the ignorance for criterion $j$. The different values $Crit_v$ have been defined empirically. The criteria discounting values used in the previous Speed Limit Assistant have been conserved for the multi-criterion fusion. The retained set of discounting values $\alpha_{MC}$ respectively applied on $C_{FC}$, $C_{RT}$, $C_C$, $C_I$ and $C_{HR}$ is:

$$\alpha_{MC} = [0.5, 1.0, 1.0, 0.5, 1.0] \tag{5.9}$$

For comparison purposes, a description of the original *SLA bba* is also provided here. The latter determines the mass over the different navigation focal speeds in three steps:

- It first determines the level of ignorance related to the navigation (cf. (5.11)). This ignorance is proportional to the focal speed number.

- The *bba* over the focal speeds is then processed using a Weighted Sum ($WS$) regarding the set of discounting coefficients presented in (5.10). By having a closer look on the discounting coefficient set, it can be seen that more importance are given to $C_4$, $C_5$ and $C_7$ so on the road type, on the city driving status and on the highway ramp information. Indeed, these coefficients are the most suited to discriminate the different focal speeds. Finally the speed belief mass is reduced by the ignorance (see (5.11)).

$$\alpha_{WS} = [0.25, 0.5, 0.5, 1.0, 1.0, 0.5, 1.0, 0.25] \tag{5.10}$$

- The *bba* over the focal speeds complementary is then processed regarding the level of ignorance and the value of the belief mass in the speed so that the sum of the different belief masses becomes 1.

In summary the *bba* process for the navigation speeds is expressed as:

$$\begin{cases} m_n(\Theta) & = N_{fs} \cdot 0.05 \\ m_n(H_n) & = \left(\frac{0.25 \cdot C_1 + 0.5 \cdot C_2 + 0.5 \cdot C_3 + C_4 + C_5 + 0.5 \cdot C_6 + 1 \cdot C_7 + 0.25 \cdot C_8}{5}\right) - m_n(\Theta) \\ m_n(H_n^c) & = 1 - m_n(H_n) - m_n(\Theta) \end{cases} \tag{5.11}$$

with $N_{fs}$ the number of focal speeds.

### 5.5.3 Confidence Variables

As shown in the previous section, the belief mass identification is obtained through the evaluation of linear expressions regarding a confidence variable. If the navigation and vision *bba* model is similar, the determination of the confidence variable is different for each sensor.

The vision confidence variable $C_{v_{vis}}$ corresponds to the confidence in the detected speed provided by the $SLSR$ ($SLSR_{confidence} \in [0, 1]$):

$$C_{v_{vis}} = SLSR_{confidence} \tag{5.12}$$

For instance, consider a high confidence in the detected speed, which corresponds to a high value for $SLSR_{confidence}$ ($> 0.5$). This involves the vision *bba* to give belief in the speed and in the ignorance (cf. (5.7)). Contrary to this, if the confidence in the detected speed is low, $SLSR_{confidence}$ will also be low ($< 0.5$) such that the vision *bba* will give belief only on the complementary speed and on the ignorance. In addition, $C_{v_{vis}}$ is reduced by a Forgiveness Factor ($FF$). This $FF$ has been used as a speed sign detected a few minutes ago may not be relevant anymore regarding the road context. The confidence given to a detected sign is consequently reduced here by 0.1 each $30s$.

Contrary to $C_{v_{vis}}$, the navigation confidence variable $C_{v_{nav}}$ is a combination of the navigation system reliability indicators which are the $HDOP$, the $MLCP$ and the *ADASAttribute*. These indicators describe the quality of the positioning, localization and the *Digital Map Database* (cf. Section.4.5.4):

$$C_{v_{nav}} = \left(1 - \left(\frac{HDOP}{HDOP_{\max}}\right)\right) \cdot \left(1 - \left(\frac{MLCP}{MLCP_{\max}}\right)\right) \cdot ADASAttribute \tag{5.13}$$

Remind that the lower the $HDOP$ and $MLCP$ values are, the higher the positioning and localization are. Both indicators are consequently compared to a maximum value, respectively $HDOP_{max}$ and $MLCP_{max}$ which represent the worst reliability level. These values have been determined empirically so that $HDOP_{max} = 40$ and $MLCP_{max} = 100000$. The *ADASAttribute* refers to a *Digital Map Database* attribute which can be only *activated* or *deactivated*. For the determination of $C_{v_{nav}}$, this activation or deactivation state have been translated into numerical values such that they respectively correspond to 0.9 and 0.7.

It is also clear that this confidence variable is conservative. Indeed, consider that $HDOP = 1$, $MLCP = 50000$ and an activated *ADASAttribute*. This example presents a situation in which two over three of the navigation system components are reliable (the positioning and the *Digital Map Database*). Nevertheless, as the localization is average, the navigation confidence variable is low ($C_{v_{nav}} = 0.44$), thus the navigation belief mass estimation will only be done on the speed complementary and on the ignorance.

### 5.5.4 Simulation Context Description

Simulation results have been obtained with sensors simulators reproducing the real-time $SLA$ behavior. A snapshot of the considered *RTMaps* diagram is presented in Fig.5.30. On this figure, two *RTMaps* block interfaces have been highlighted: the interface of the block containing the sensors simulation data and the block dedicated to the multi-level fusion which are respectively marked by 1 and 2. A closer look on the first interface shows that the simulator allows the user to define all the required information for the multi-level fusion. Indeed, it helps to define: the vision speed (*SpeedLimitVision*), its confidence (*ConfVision*), the HDOP (*HDOP*), the MLCP (*MLCP Probability*), the navigation speed (*SpeedLimitNav*), the road type (*RoadType*),

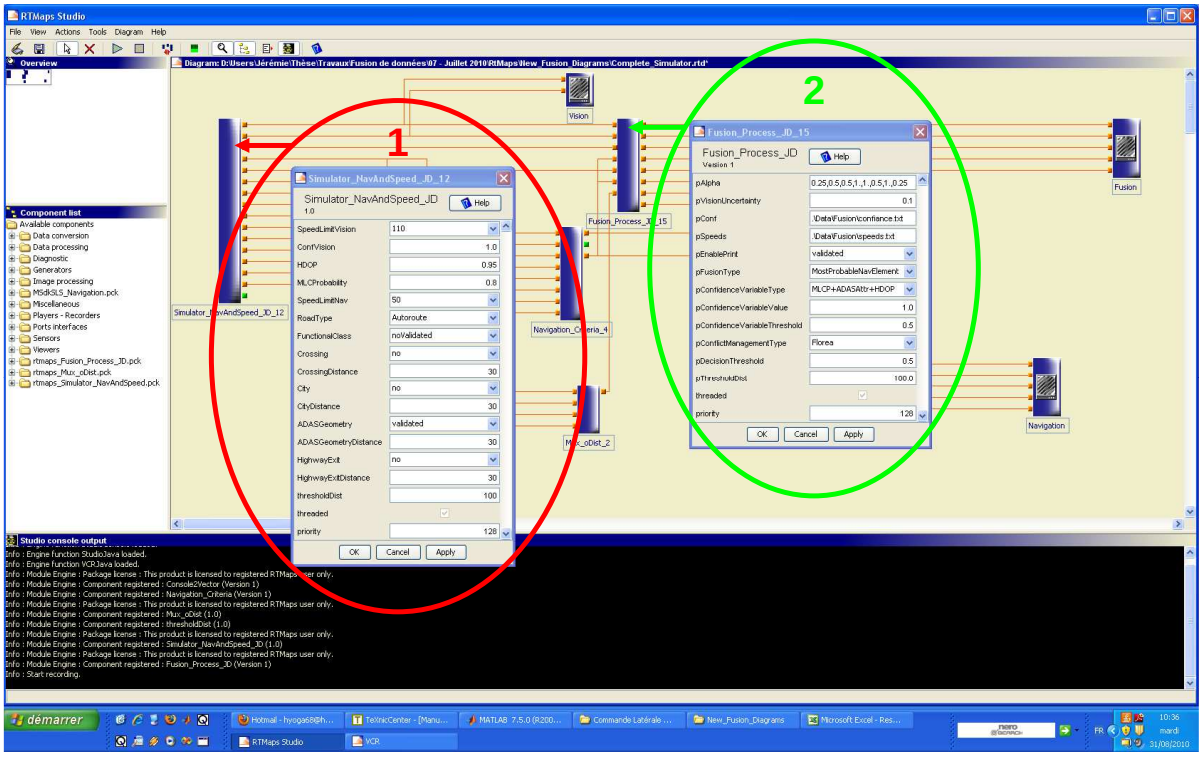



Figure 5.30: Simulation Conditions

the functional class (*FunctionalClass*), the presence of an intersection (*Crossing*), the current city status (*City*), the ADAS Attribute status (*ADASGeometry*), and the presence of a highway ramp (*HighwayExit*). Also note that some of the information i.e. the presence of an intersection has a configurable distance which is not used in the current fusion process.

The second interface, related to the multi-level fusion presents the fusion conditions: the multi-sensor fusion is processed between one navigation and one vision speed, the most probable ones (*pFusionType*) and the navigation confidence variable $C_{v_{nav}}$ is proportional to the $HDOP$, the $MLCP$ and the $ADAS\ Attribute$ values (*pConfidenceVariableType*). The confidence variable threshold $\tau$ is fixed to 0.5 (*pConfidenceVariableThreshold*) and the conflict redistribution is based on *Florea*'s redistribution operator (*pConflictManagementType*). Finally, the *Decision* step of the raw multi-sensor fusion selects a speed only if it possesses a belief mass at least equal to 0.5 (*pDecisionThreshold*). This configuration corresponds to the strategy presented in Chapter.4. This simulator allows quick manual modifications of the different fusion variables, consequently helps to simulate real-time fusion considering specific road contexts.

### 5.5.5 Multi-criterion Fusion Validation

The multi-criterion fusion represents the first level of the multi-level fusion. The main interest is to proceed, in a first step, to a fusion of several criteria from the navigation, in order to evaluate the reliability of the extracted data (i.e. the speed limit and the road context indicators), and detect the potential erroneous/contradictory information. The results of this $SLA$ are here compared to the results of the Speed Limit Assistant available in [Bradai, 2007]. This comparison is done regarding three different scenarios: a situation in which coherent criteria w.r.t. the situation is considered and a situation in which incoherent criteria w.r.t. the situation are

Table 5.6: Road Context Configuration for Coherent Navigation Information

| *Navigation Attributes* | *Weighted Sum Interpretation* | *Multi-criterion Interpretation* |
|---|---|---|
| Navigation Speed = 50 | FS: 30, 70, 100, 110, 120, 130, *unlimited* | FS: 30, 70, 100, 110, 120, 130, *unlimited* |
| MLCP = 10000 | $C_1 = 0.9$ | MLCP = 10000 |
| ADAS Attribute Validated | $C_2 = 0.9$ | ADASAttribute = 0.9 |
| Functional Class = 4 | $C_3 = 0.7$ | $C_{FC} = 0.7$ |
| Communal Road | $C_4$ is *Communal* | $C_{RT}$ is *Communal* |
| In-city Driving | $C_5$ is *In-city* | $C_C$ is *In-city* |
| No Intersection | $C_6$ is *No Intersection Detected* | $C_I$ is *No Intersection Detected* |
| No Highway Ramp | $C_7$ is *No Highway Ramp Detected* | $C_{HR}$ is *No Highway Ramp Detected* |
| Guidance Mode Validated | $C_8 = 0.9$ | No impact |
| HDOP = 1 | No impact | HDOP = 1 |

considered. Both of these scenarios are based on reliant navigation information, contrary to the third situation in which the navigation information is considered to be unreliable.

#### 5.5.5.1 Coherent Navigation Information

Here, the comparison of both navigation *bba* approaches (weighted sum versus multi-criterion) is carried out in a configuration in which the navigation criteria are coherent with the real-road context. Let consider the road context given in Table.5.6. This situation corresponds to a classical in-city driving as the navigation speed is $50km.h^{-1}$, thus implies the consideration of 30, 70, 100, 110, 120, 130 and *unlimited* as focal elements (cf. Table.D.1) for both approaches. Then the *Vehicle* is driving on an in-city communal road with a functional class of 4 (describing a low importance road) but without the presence of any intersection or highway ramp. The positioning, the localization and the digital map are here considered as accurate: $HDOP = 1$, $MLCP = 10000$, *ADAS Attribute* is validated[28]. Consequently, the navigation belief masses over the focal speeds should be high. Finally, the guidance mode ($C_8$) which is only considered in the weighted sum approach, is activated.

**Weighted Sum Results** Considering the road configuration presented in this table, the navigation *bba* of each focal element have been calculated as follows (cf. (5.11)):

$$\begin{aligned}
&m_{30}(\Theta) && = 7 \cdot 0.05 && = 0.35\\
&m_{30}(30) && = 0.63 - 0.35 && = 0.28\\
&m_{30}(30^c) && = 1 - 0.28 - 0.35 && = 0.37\\
&\vdots && && \\
&m_{unlimited}(\Theta) && = 7 \cdot 0.05 && = 0.35\\
&m_{unlimited}(unlimited) && = 0.48 - 0.35 && = 0.13\\
&m_{unlimited}(unlimited^c) && = 1 - 0.13 - 0.35 && = 0.52
\end{aligned} \tag{5.14}$$

The resulting *bba* are presented in Fig.5.31. This $SLA$ generates masses on the speeds $m_j(H_j)$, on their complementary $m_j(H_j^c)$ and the ignorance $m_j(\Theta)$. This allows to determine the level of confidence of the propositions: *"It is this speed"*, *"It is not this speed"*, and *"One ignore if it is this speed or not"*. Nevertheless, on this figure it can be clearly seen that the $SLA$ is undecided about the right speed limit as no proposition possesses a belief mass higher than

[28]Remind that the $HDOP$ and the $MLCP$ are compared to their maximum value. In this example $MLCP = 10000$, thus $1 - \frac{MLCP}{MLCP_{max}} = 0.8$. This implies $C_1 = 0.9$.

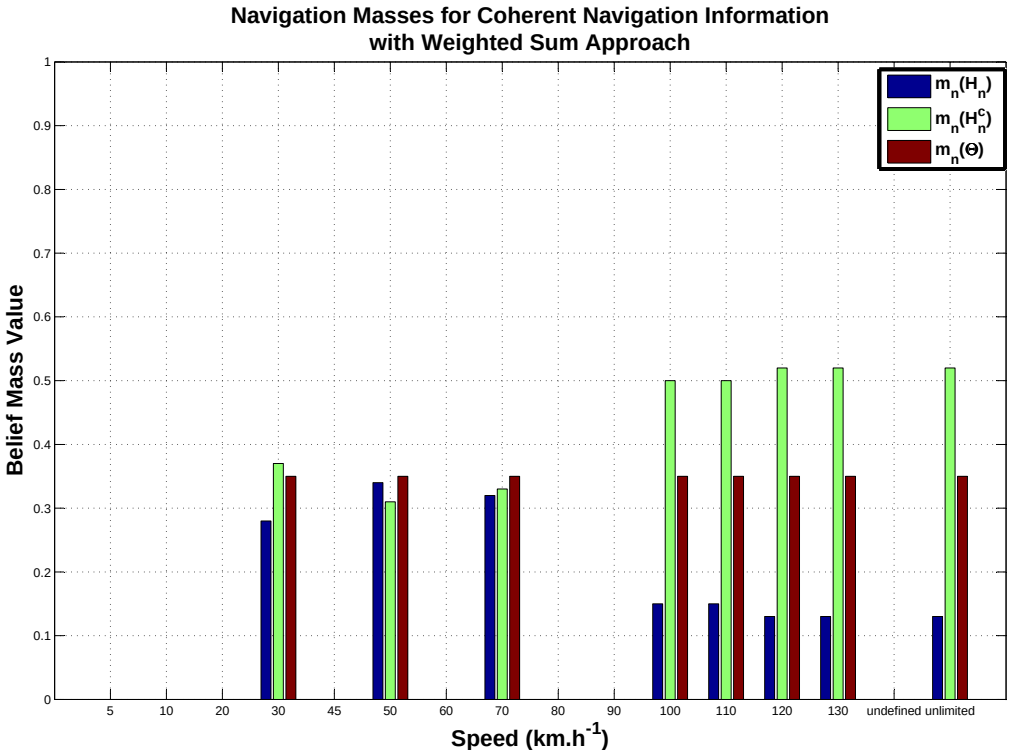



Figure 5.31: SLA Navigation Mass Estimation Using Weighted Sum

its complementary and its ignorance at the same time. This occurs even if the road context described by the criteria is coherent with the speed extracted from the database. In addition, $50km.h^{-1}$ has a belief mass of only 0.34, a belief mass on its complementary of 0.31 and an ignorance of 0.35, so is defined by values which are very close. This observation is also true for $30km.h^{-1}$ and $70km.h^{-1}$. This low belief generation mainly originates from the high ignorance value of 0.35 which is due to the large number of focal speeds. Nevertheless it is important to note that high speeds which are not coherent with the current context (100, 110, 120, 130 and *unlimited*) are rejected as they have similar and very small confidences of 0.15 with strong complementary masses of 0.5. Consequently, if the Speed Limit Assistant cannot determine the best limit speed, it is confident in the fact that it could not be a high speed.

**Multi-criterion Results** The calculation of the focal speeds *bba* firstly requires the determination of the confidence variable $C_{v_{nav}}$ defined by (4.26) and here equal to 0.68.

This value indicates that the *SLA* is quite confident in the data provided by the navigation system. The criteria *bba* are then calculated based on the linear expressions given in (5.8). As $C_{v_{nav}} > \tau = 0.5$, the calculation of these masses is obtained for $50km.h^{-1}$ as following:

$$\begin{array}{ll} m_{C_{FC,50}}(50) & = 0.25 \\ m_{C_{FC,50}}(50^c) & = 0 \\ m_{C_{FC,50}}(\Theta) & = 0.75 \\ \vdots & \\ & \\ m_{C_{HR,50}}(50) & = 0.04 \\ m_{C_{HR,50}}(50^c) & = 0 \\ m_{C_{HR,50}}(\Theta) & = 0.96 \end{array} \tag{5.15}$$

The next step consist in the discounting resulting in the *bba* $\hat{m}_{j,n}(H_n)$, $\hat{m}_{j,n}(H_n^c)$ and $\hat{m}_{j,n}(\Theta)$ which are then used by *Dempster*'s conjunctive combination operator based on the multi-criterion rules available in (4.31). For $50km.h^{-1}$, this combination results in:

$$\begin{array}{lcll} m_{FC...HR,50}(50) & = & \prod_{j=FC}^{HR}(1-m_{j,i}(H_i^c)) - \prod_{j=FC}^{HR} m_{j,i}(\Theta) & = 0.67 \\ m_{FC...HR,50}(50^c) & = & \prod_{j=FC}^{HR}(1-m_{j,i}(H_i)) - \prod_{j=FC}^{HR} m_{j,i}(\Theta) & = 0.0 \\ m_{FC...HR,50}(\Theta) & = & \prod_{j=FC}^{HR} m_{j,i}(\Theta) & = 0.33 \end{array} \tag{5.16}$$

Finally, the multi-criterion fusion on the 7 focal speeds results in the situation presented in Fig.5.32. First, it can be noticed that the multi-criterion approach generates mass only on the different speeds ($m_n(H)$) and ignorance ($m_n(\Theta)$). This is due to the considered *bba* model excluding the generation of mass over an hypothesis and over its complementary simultaneously. Moreover, due to the high value of the confidence variable ($C_{v_{nav}} = 0.68$), the multi-criteria fusion believes in the speeds $H_n$ but not on the speed complementaries $H_n^c$. Then, this approach greatly improves the readability of the results. Indeed, if the weighted sum gives low and close beliefs in low speeds, the multi-criterion approach results in strong beliefs in theses speeds as $m_{30}(30) = 0.61$, $m_{50}(50) = 0.67$ and $m_{70}(70) = 0.64$ with low ignorance (respectively 0.39, 0.33 and 0.36). On the opposite, high speeds are more subject to ignorance, so are not strongly rejected as for the weighted sum approach. This simulation proves that the multi-criterion provides beliefs of higher importance in the navigation speed which is coherent with the road context, than the previous approach based on weighted sum.

#### 5.5.5.2 Incoherent Navigation Information

Lets now consider the configuration described by Table.5.7. This table defines a road context which possesses similar characteristics than the previous one. Indeed, the navigation speed is still $50km.h^{-1}$, so implies the consideration of 30, 70, 100, 110, 120, 130 and *unlimited* as focal elements. Then, the positioning, the localization and the digital map are here also considered as accurate as $HDOP = 1$, $MLCP = 10000$ and as the *ADAS Attribute* is validated. There are still no intersection or highway ramp detection and finally, the guidance mode is still activated.

However, the main change regarding the preceding road context is that the *Vehicle* is said to drive on a highway outside an urban area. This is incoherent with the navigation speed of $50km.h^{-1}$ usually referring to communal or departmental in-city roads. The speed extracted from the database is consequently not coherent with this road context.

Results of the mass estimation using the weighted sum and the multi-criterion approaches are respectively presented in Fig.5.33 and in Fig.5.34. Both approaches reject the navigation

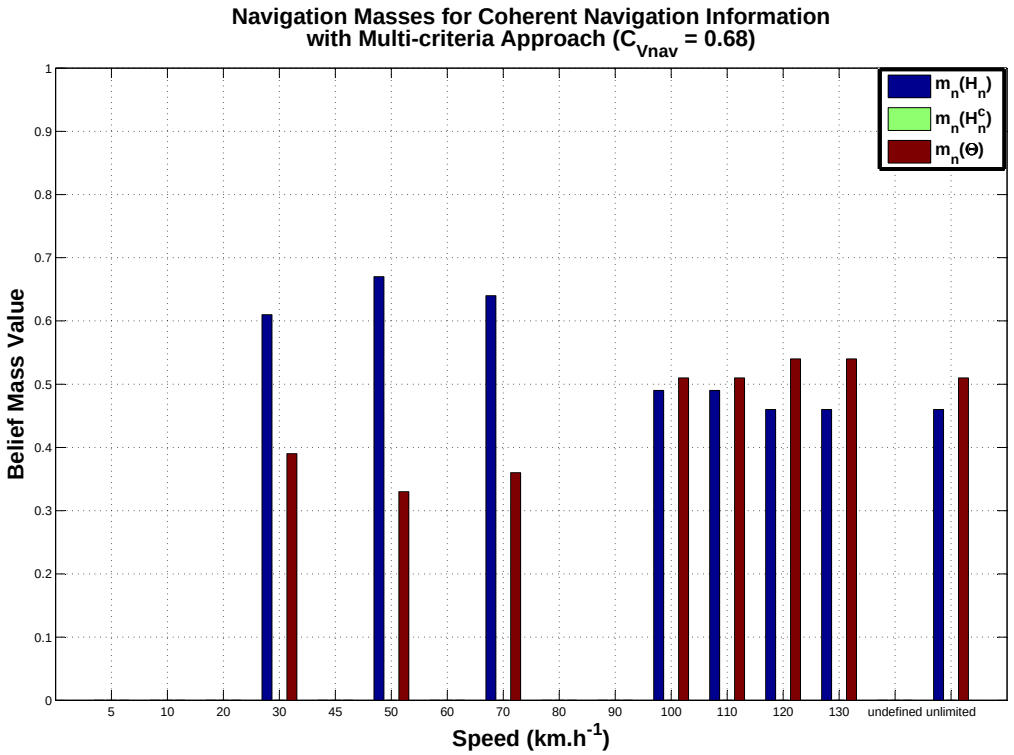



Figure 5.32: SLA Navigation Mass Estimation Using Multi-criterion Fusion

Table 5.7: Road Context Configuration for Incoherent Navigation Information

| *Navigation Attributes* | *Weighted Sum Interpretation* | *Multi-criterion Interpretation* |
|---|---|---|
| Navigation Speed = 50 | FS: 30, 70, 100, 110, 120, 130, *unlimited* | FS: 30, 70, 100, 110, 120, 130, *unlimited* |
| MLCP = 10000 | $C_1 = 0.9$ | MLCP = 10000 |
| ADAS Attribute Validated | $C_2 = 0.9$ | ADASAttribute = 0.9 |
| Functional Class = 4 | $C_3 = 0.7$ | $C_{FC} = 0.7$ |
| Highway | $C_4$ is *Highway* | $C_{RT}$ is *Highway* |
| Out-city Driving | $C_5$ is *Out-city* | $C_C$ is *Out-city* |
| No Intersection | $C_6$ is *No Intersection Detected* | $C_I$ is *No Intersection Detected* |
| No Highway Ramp | $C_7$ is *No Intersection Detected* | $C_{HR}$ is *No Highway Ramp Detected* |
| Guidance Mode Validated | $C_8 = 0.9$ | No impact |
| HDOP = 1 | No impact | HDOP = 1 |

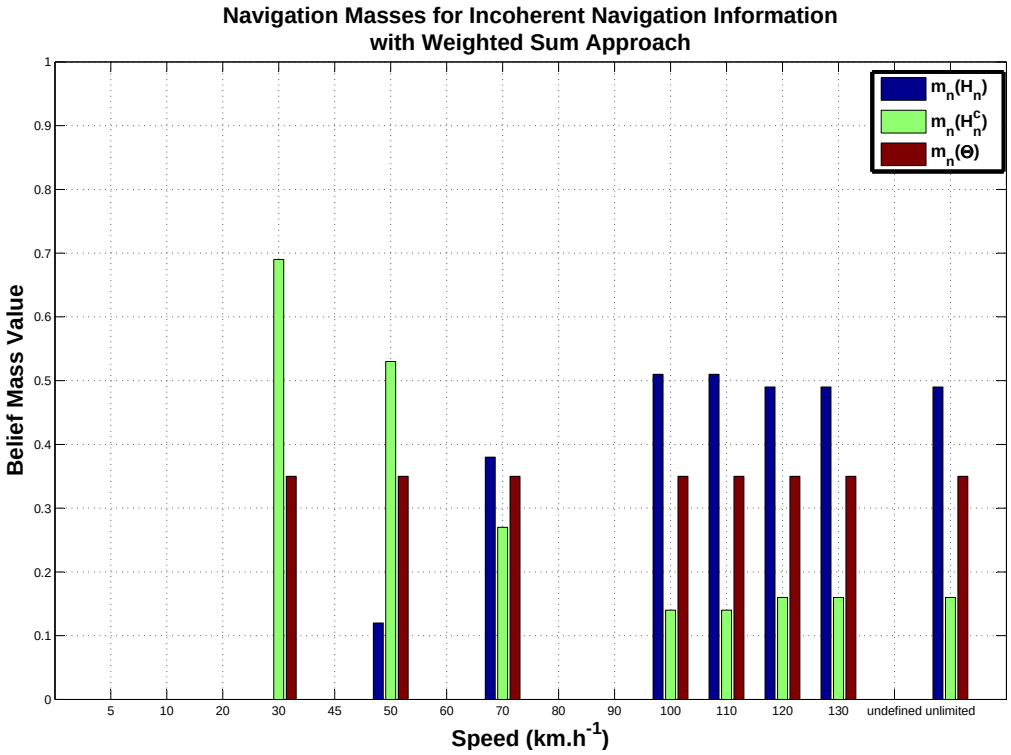



Figure 5.33: SLA Navigation Mass Estimation Using Weighted Sum

speed ($50km.h^{-1}$) as the weighted sum generates a high complementary mass $m_{50}(50^c) = 0.53$ with simultaneously a low belief in the given speed $m_{50}(50) = 0.12$ and as the multi-criterion fusion results in high ignorance $m_{50}(50^c) = 0.53$ with an average belief mass $m_{50}(50) = 0.47$. For this specific configuration, both approaches were consequently able to detect the navigation incoherency. More generally, both approaches have favorite high speeds. Indeed, $30km.h^{-1}$ has been completely rejected (the weighted sum giving a belief of 0.0), while $100km.h^{-1}$, $110km.h^{-1}$, $120km.h^{-1}$, $130km.h^{-1}$ and *unlimited* present similar belief values. However, the level of these high speed beliefs is different regarding the adopted approach. Indeed, the weighted sum gives an average belief of 0.5 to them, while the multi-criterion fusion believes more in them (around 0.76). As for the previous test, the weighted sum approach is not strongly enforcing the belief over the speeds which are coherent with the road context (described by the criteria) due to the high level of ignorance. This example again reveals the necessity of multi-criterion fusion which generates the different masses independently of the number of focal elements.

#### 5.5.5.3 Low Navigation Information Reliability

Let's now consider the road context given in Table.5.8 corresponding to a normal highway driving situation as the navigation speed is $130km.h^{-1}$. This implies the consideration of 50, 70, 90, 110 and $130km.h^{-1}$ as focal elements (cf. Table.D.1). Then, the *Vehicle* is driving on an out-city highway with a functional class of 0 (so describes a very high importance road), without the presence of any intersection or highway ramp. The navigation information is of average quality: $MLCP = 25000$, $HDOP = 10$ and *ADAS Attribute* is not activated[29]. Finally, the guidance

[29] $1 - \frac{MLCP}{MLCP_{max}} = 0.5$. This implies $C_1 = 0.5$.

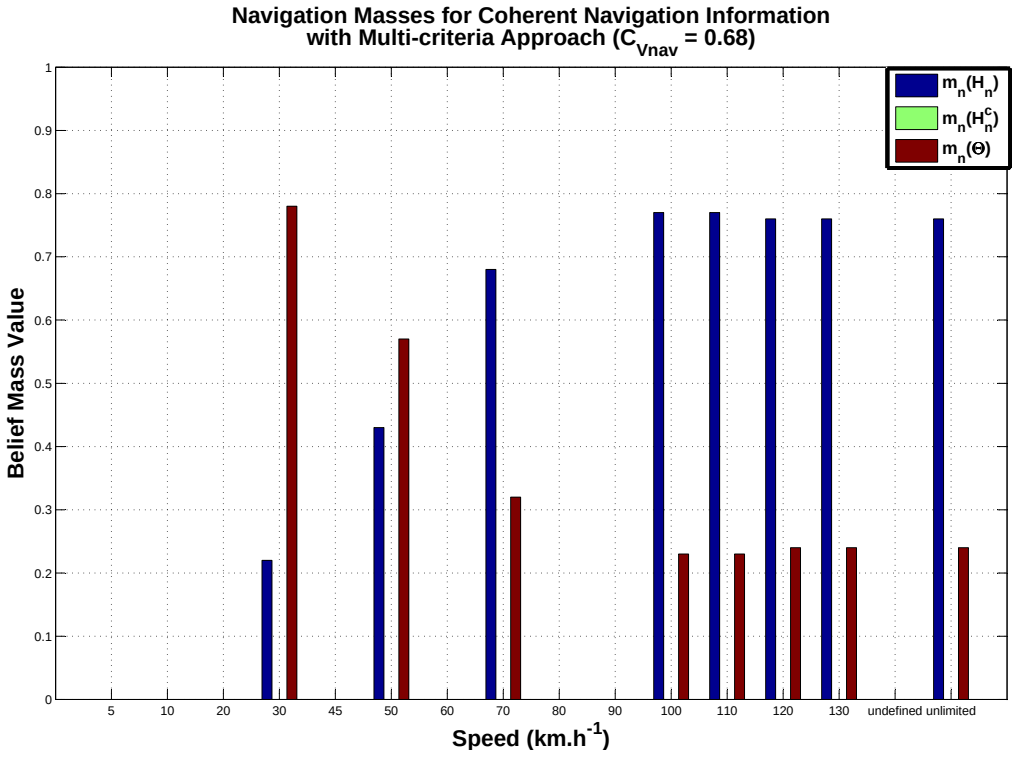



Figure 5.34: SLA Navigation Mass Estimation Using Multi-criterion Fusion

mode is activated.

Considering this configuration, the calculation of the belief masses using the multi-criterion fusion is processed with a low navigation confidence variable $C_{v_{nav}}$ equal to 0.18.

As $C_{v_{nav}} < \tau = 0.5$, only non-null *bba* will be generated for $H^c$ and $\Theta$.

$$\begin{aligned} m_{C_{FC,50}}(50) &= 0 \\ m_{C_{FC,50}}(50^c) &= 0.58 \\ m_{C_{FC,50}}(\Theta) &= 0.42 \\ &\vdots \end{aligned} \tag{5.17}$$

Table 5.8: Road Context Configuration for Coherent Navigation Information

| *Navigation Attributes* | *Weighted Sum Interpretation* | *Multi-criterion Interpretation* |
|---|---|---|
| Navigation Speed = 130 | FS: 50, 70, 90, 110 | FS: 50, 70, 90, 110 |
| MLCP = 25000 | $C_1 = 0.5$ | MLCP = 25000 |
| ADAS Attribute Not Validated | $C_2 = 0.7$ | ADASAttribute = 0.7 |
| Functional Class = 0 | $C_3 = 0.9$ | $C_{FC} = 0.9$ |
| HigHway Road | $C_4$ is *HigHway* | $C_{RT}$ is *HigHway* |
| Out-city Driving | $C_5$ is *Out-city* | $C_C$ is *Out-city* |
| No Intersection | $C_6$ is *No Intersection Detected* | $C_I$ is *No Intersection Detected* |
| No Highway Ramp | $C_7$ is *No Highway Ramp Detected* | $C_{HR}$ is *No Highway Ramp Detected* |
| Guidance Mode Validated | $C_8 = 0.9$ | No impact |
| HDOP = 10 | No impact | HDOP = 10 |

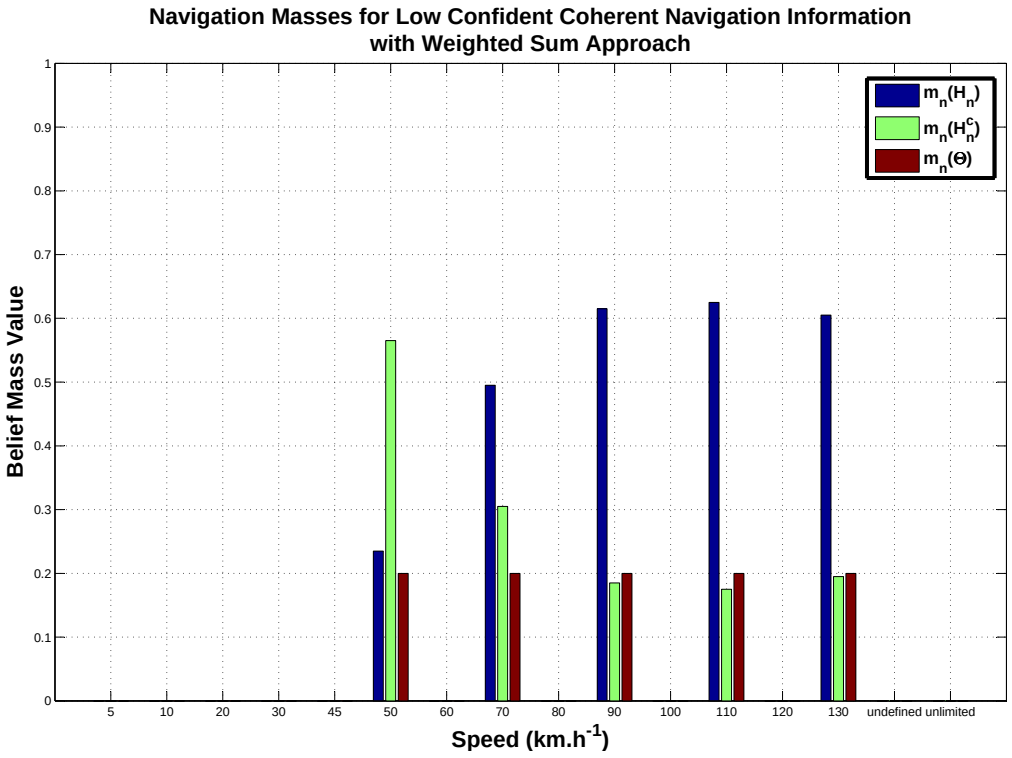



Figure 5.35: SLA Navigation Mass Estimation Using Weighted Sum

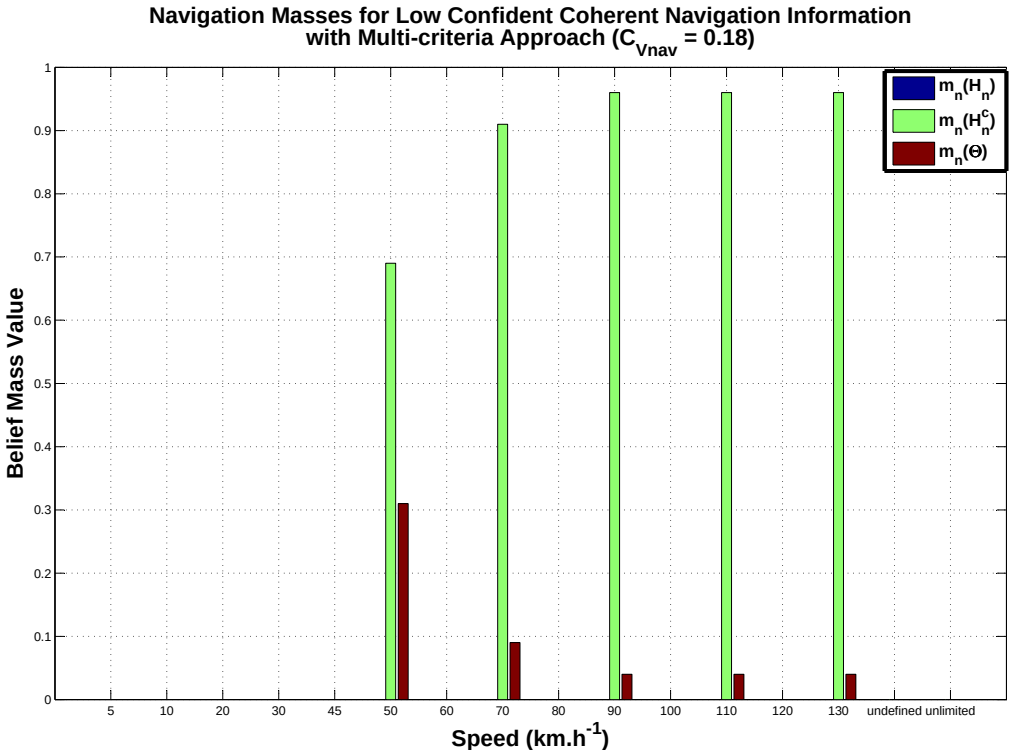



Figure 5.36: SLA Navigation Mass Estimation Using Multi-criterion Fusion

The results of the weighted sum and multi-criterion approaches are presented respectively in Fig.5.35 and Fig.5.36. The multi-criterion approach only generates non null belief masses on the complementary propositions and the ignorance. This contrasts with the preceding tests which were only generating masses on the propositions and the ignorance. This is due to the very low value of the navigation confidence variable ($C_{v_{nav}} = 0.18$), i.e. low $MLCP$, low $HDOP$ and *ADAS Attribute* deactivated. Indeed, the positioning, localization and digital map may be subject to large inaccuracies. Consequently, the navigation information cannot be strongly trusted. This point has been correctly integrated by the multi-criterion approach as it rejects the focal speeds with very high *bba* on the complementary hypothesis (around 0.95). On the opposite, the weighted sum approach presents strong belief masses in the navigation information regardless to its quality: high speeds ($90km.h^{-1}$, $110km.h^{-1}$ and $130km.h^{-1}$) have belief masses which are higher than 0.6. The weighted sum approach is not able to integrate correctly the reliability of the navigation information into the belief mass calculation.

### 5.5.6 Multi-sensor Fusion Validation

The multi-sensor fusion (fusion between the fused criteria of the navigation and the vision information) represents the second level of the multi-level fusion. The enhancements brought to this fusion step is the reconsideration of the importance given to each sensor. Indeed, the previous approach was always looking for a navigation speed (focal speed) matching the vision speed, so was giving more weight to the vision information. Contrary to this, the new approach takes the best information from each sensor even if they are not concordant, thus gives equal importance to both sensors. Furthermore the automatic conflict redistribution is no more applied as the conflict is now considered as an information for the determination of the final speed limit. The benefits brought by these enhancements are shown here, as for the previous section, through the comparison of both approaches.

#### 5.5.6.1 Concordant Sensor Information

The first simulation is carried out using the configuration presented in Table.5.6, so in a case in which the navigation speed is coherent with the criteria (describing the road context). It has been shown that in this case, the multi-criterion approach was giving better results. Indeed, the weighted sum, due to the high ignorance was not able to select accurately a speed even if it favorites low speeds. Contrary to this, the multi-criterion fusion was strongly enforcing low speeds with higher level of belief and, with the selection of the maximum of belief, was able to select $50km.h^{-1}$ as the final navigation speed.

To process the multi-sensor fusion, the vision information *bba* have to be calculated. In fact, as mentioned in Section.4.5.5, the weighted sum $SLA$ considers the $SLSR$ information as the vision masses directly. This approach has been modified so that the vision *bba* are now determined using the *Rombaut/Gruyer* model (5.7). Lets consider the information provided by the camera to be as presented in Table.5.9. From this table, the difference of belief mass estimation can be depicted.

This vision information is then fused with the navigation information obtained in Section.5.5.5.1. Whatever the results of the navigation *bba* over the different focal speeds, as the vision speed equals to the navigation speed, the weighted sum $SLA$ combines the navigation mass attributed to $50km.h^{-1}$ with the corresponding masses. This is not the case for the multi-level fusion as it first selects the navigation speed which has the maximum of belief. In this case, this speed is also $50km.h^{-1}$. Consequently, for both approaches, the navigation and vision speeds

Table 5.9: Vision Information to be Fused with Coherent Navigation Information

| *Vision Attributes* | *Weighted Sum Interpretation* | *Multi-level Interpretation* |
|---|---|---|
| Detected Speed = 50 | Vision Speed = 50 | Vision Speed = 50 |
| Confidence in Detection = 0.90 | $m_v^{WS}(50) = 0.90$<br>$m_v^{WS}(50^c) = 0.00$<br>$m_v^{WS}(\Theta) = 0.10$ | $m_v^{ML}(50) = 0.80$<br>$m_v^{ML}(50^c) = 0.00$<br>$m_v^{ML}(\Theta) = 0.20$ |

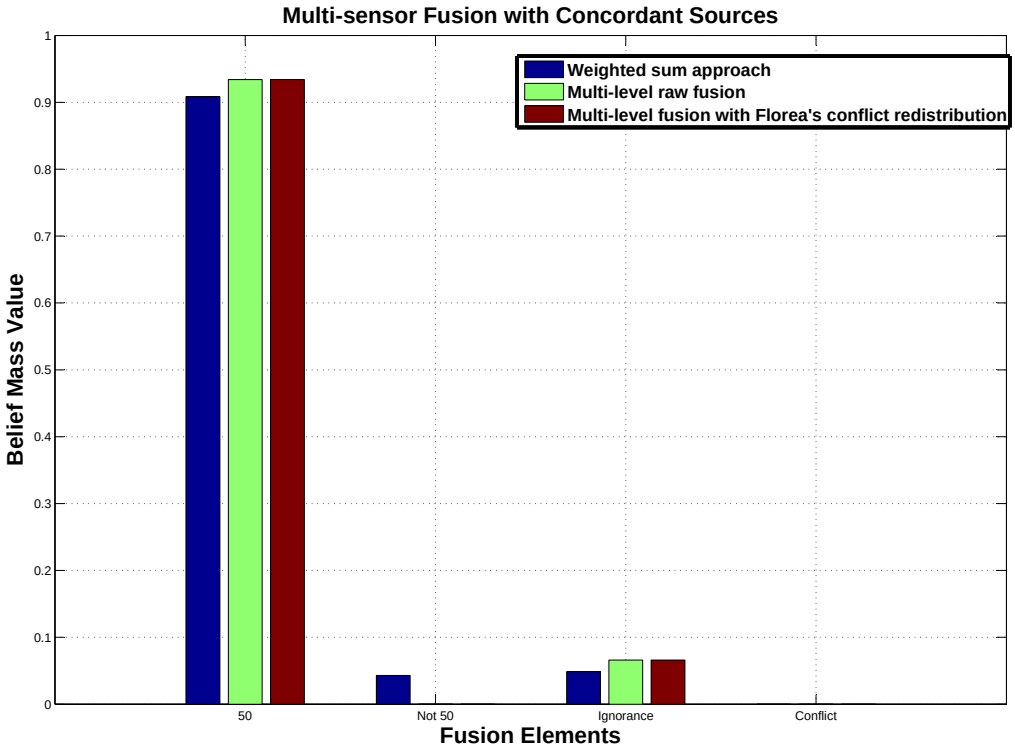



Figure 5.37: Multi-sensor Fusion Results for Concordant Sensor Information

are similar. The multi-sensor fusion *bba* using the Weighted Sum ($WS$) and the Multi-Level ($ML$) are respectively presented in (5.18) and (5.19). Remind that the multi-sensor fusion based on weighted sum was normalizing the conflict, the conflict presented in (5.18) is consequently redistributed to the other elements so that $m_{vn}^{WS}(\emptyset) = 0$.

$$\begin{array}{lll} m_{vn}^{WS}(50) & = \frac{1}{1-m_{vn}^{WS}(\emptyset)} \cdot \left(m_v^{WS}(50) \cdot \left(m_n^{WS}(\Theta) + m_n^{WS}(50)\right) + m_v^{WS}(\Theta) \cdot m_n^{WS}(50)\right) & = 0.91 \\ m_{vn}^{WS}(50^c) & = \frac{1}{1-m_{vn}^{WS}(\emptyset)} \cdot \left(m_v^{WS}(50^c) \cdot \left(m_n^{WS}(\Theta) + m_n^{WS}(50^c)\right) + m_v^{WS}(\Theta) \cdot m_n^{WS}(50^c)\right) & = 0.04 \\ m_{vn}^{WS}(\Theta) & = \frac{1}{1-m_{vn}^{WS}(\emptyset)} \cdot m_v^{WS}(\Theta) \cdot m_n^{WS}(\Theta) & = 0.05 \end{array} \tag{5.18}$$

$$\begin{array}{lll} m_{vn}^{ML}(\emptyset) & = m_v^{ML}(50) \cdot m_n^{ML}(50^c) + m_v^{ML}(50^c) \cdot m_n^{ML}(50) & = 0.0 \\ m_{vn}^{ML}(50) & = m_v^{ML}(50) \cdot \left(m_n^{ML}(\Theta) + m_n^{ML}(50)\right) + m_v^{ML}(\Theta) \cdot m_n^{ML}(50) & = 0.93 \\ m_{vn}^{ML}(50^c) & = m_v^{ML}(50^c) \cdot \left(m_n^{ML}(\Theta) + m_n^{ML}(50^c)\right) + m_v^{ML}(\Theta) \cdot m_n^{ML}(50^c) & = 0.0 \\ m_{vn}^{ML}(\Theta) & = m_v^{ML}(\Theta) \cdot m_n^{ML}(\Theta) & = 0.07 \end{array} \tag{5.19}$$

Results of the multi-sensor fusion are available in Fig.5.37. This figure clearly shows that for both Speed Limit Assistants, the fusion results in a large belief in $50km.h^{-1}$. This is quite

Table 5.10: Vision Information to be Fused with Coherent Navigation Information

| *Vision Attributes* | *Weighted Sum Interpretation* | *Multi-level Interpretation* |
|---|---|---|
| Detected Speed = 110 | Vision Speed = 110 | Vision Speed = 110 |
| Confidence in Detection = 0.90 | $m_v^{WS}(110) = 0.90$<br>$m_v^{WS}(110^c) = 0.00$<br>$m_v^{WS}(\Theta) = 0.10$ | $m_v^{ML}(110) = 0.80$<br>$m_v^{ML}(110^c) = 0.00$<br>$m_v^{ML}(\Theta) = 0.20$ |

obvious as both sensors are opting for this speed. Nevertheless, a few differences can already be stressed: the multi-level approach only generates mass on the speed and on the ignorance when both sensors are concordant contrary to the weighted sum based approach which generates a small belief mass on the speed complementary ($50^c$). Then, remind that the new approach provides two speed limits to the *Driver*: the raw speed limit obtained from the raw multi-sensor fusion and an indicative speed limit obtained after the conflict redistribution using *Florea*'s operator. The latter, as there is no conflict generation, has no impact so that the fusion with conflict redistribution gives the same results than the raw fusion.

#### 5.5.6.2 Conflict Between Sensors

Lets now consider a second configuration in which the vision is not giving the same speed than the navigation. This can be obtained by changing the vision information provided by the camera into the values presented in Table.5.10. The navigation information described by Table.5.6 and used in the previous section has been conserved.

Here again the vision information is then fused with the navigation information obtained in Section.5.5.5.1. However, the behavior of the weighted sum-based approach is different. Indeed, the navigation and the vision speeds are different, the system therefore looks for a focal element of the navigation which is equal to the speed defined by the vision. Since the speed $110km.h^{-1}$ selected by the vision is a focal element of $50km.h^{-1}$, the fusion is then processed for $110km.h^{-1}$. For the multi-level approach, the fusion is processed between the most confident navigation speed and the vision speed. Regarding to the multi-criterion fusion results presented in Section.5.5.5.1, the maximum of belief is obtained for $50km.h^{-1}$. The multi-sensor fusion is then processed for the sources which are not giving the same speeds such a presented in (4.38).

Both combination results are presented in Fig.5.38. This figure shows the combined belief mass on the Vision Speed, the Navigation Speed, the Ignorance and the Conflict. It is clear that the weighted sum-based approach has a strong belief in the Vision speed (0.85). This is mainly due to the consideration of the $110km.h^{-1}$ navigation masses and to the conflict normalization using *Dempster*'s method. The Decision step finally considers $110km.h^{-1}$ for the final speed limit. This is quite incoherent with the navigation criteria saying that the *Vehicle* is In-city and on a Communal road. Contrary to this, the multi-level approach generates small beliefs in both navigation and vision speeds. Indeed, as sensors disagree, there is a high conflict (0.54) which means that sensors provide contradictory data (considering the exhaustiveness of the discernment frame). In this particular case, before conflict management the multi-level approach is not able to take a decision based on the raw fusion. Indeed, as mentioned in Section.4.7 the raw fusion *Decision* is based on the selection of the speed having the maximum of belief considering a minimal threshold of 0.5. Here none of the speeds has a belief higher than 0.5 due to the high conflict (0.54). Consequently, regarding this *Decision* strategy, the final speed is *undefined*. The conflict redistribution (using *Florea*'s operator) involves an increase of each belief, which are then both closer to 0.5. However, as $110km.h^{-1}$ has the highest belief, it is considered as the final speed limit. These results clearly show that the automatic conflict redistribution may not

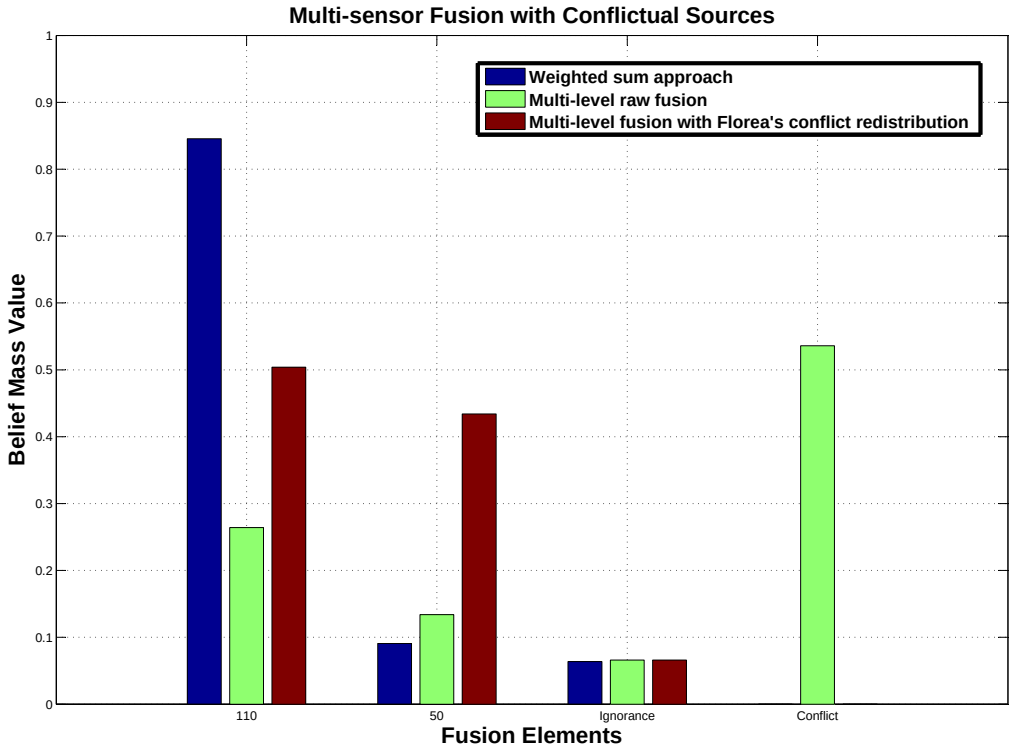



Figure 5.38: Multi-sensor Fusion Results for Contradictory Sensor Information

be adapted for this configuration as it completely masks the initial high conflict between the sources.

#### 5.5.6.3 Incoherent Navigation Information

The last simulation tests refer to the configuration tested in Section.5.5.5.2 in which the navigation information was incoherent. The benefits brought by the multi-criterion fusion were then proved as this approach was able to determine a speed limit which was much more coherent regarding the criteria. In this section, two cases are presented, the first one in which sensors agree (gives a similar initial speed) and the second one in which sensors disagree (initial speeds are not similar).

For the first test, lets consider the navigation and vision configuration to be as presented respectively in Table.5.7 and in Table.5.9. Results of both $SLAs$ are presented in Fig.5.39.

As for the previous tests, the weighted sum-based approach takes the navigation information suiting the vision speed. Consequently, it uses the navigation information calculated for $50km.h^{-1}$ and combines it to the vision information. Even if the weighted sum approach was able to detect the *Digital Map Database* incoherency, the detection is useless as the considered navigation speed has to match the vision speed. This, in addition to the conflict normalization, explains why there is a so high confidence in $50km.h^{-1}$ after the multi-sensor fusion (0.83). Contrary to this result, the multi-level approach, through the multi-criterion fusion, was able to detect the navigation incoherency, thus considered the navigation information obtained for $110km.h^{-1}$ as it possesses the maximum of belief. This obviously involves the generation of a conflict mass (of 0.62) as the sensors disagree. Consequently, as for the previous test configuration, the multi-level approach does not give a final speed limit (speed limit is *undefined*).

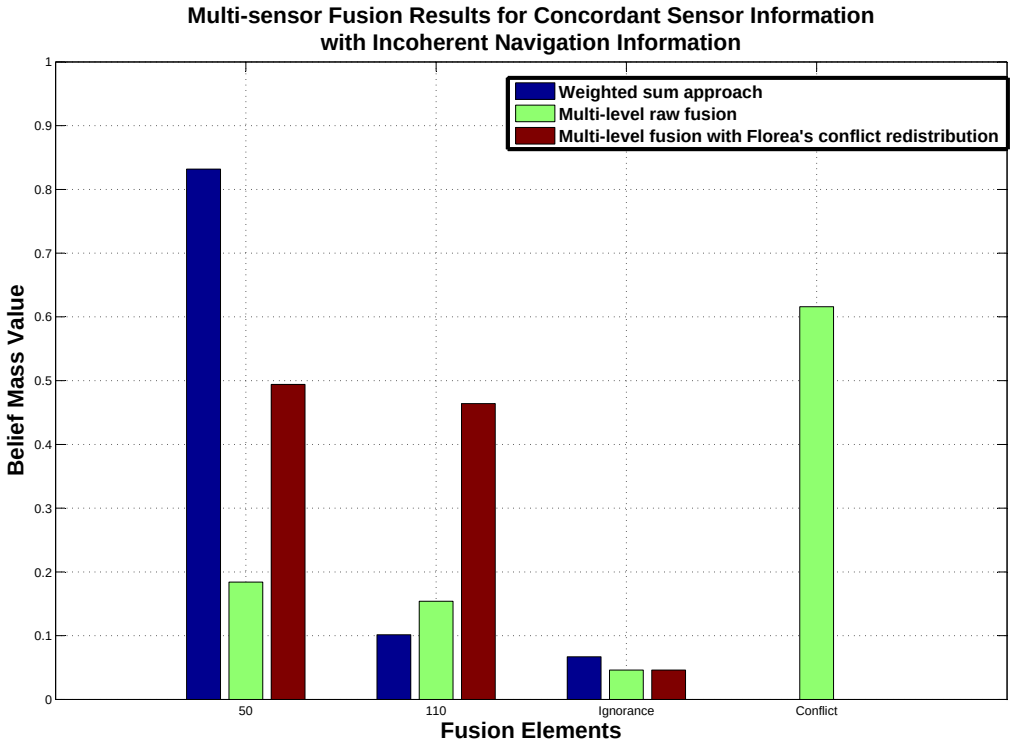



Figure 5.39: Multi-sensor Fusion Results for Concordant Sensor Information with Incoherent Navigation Information

Nevertheless, it can be remarked that the belief mass of $50km.h^{-1}$ and $110km.h^{-1}$ speeds are low but close (respectively 0.18 and 0.15) which means that it is difficult to determine the best speeds. This is also true after the conflict redistribution as the masses become respectively 0.49 and 0.46. Anyway, the vision speed is then retained for the indicative speed limit as it possessed the maximum of belief after conflict redistribution. However, as their belief mass are close, a little reduction in the confidence of the speed sign detection or a little increase of the confidence variable value may involve the consideration of $110km.h^{-1}$ as the indicative speed limit.

For the second test, lets consider the navigation and vision configuration to be as presented respectively in Table.5.7 and in Table.5.10. Results of both $SLAs$ are presented in Fig.5.40.

From this figure, it can be noted that, as usual, the belief in the vision information is very strong (0.94) for weighted sum-based approach due to the consideration of the navigation speed which matches the vision speed and to the conflict normalization. However, contrary to the previous tests, the multi-level approach also gives a large belief to $110km.h^{-1}$ (0.96). Indeed, the multi-criterion fusion was able to detect the navigation false information, thus was able to determine that regarding the criteria values, $110km.h^{-1}$ is the speed having the maximum of belief. This copes with the previous approach which considered $110km.h^{-1}$ only for matching the vision speed. Anyway, this imply the multi-sensor fusion to be processed for sensors who agree. In this case, there is almost no doubt that $110km.h^{-1}$ is the real speed limit and there is no conflict generation.

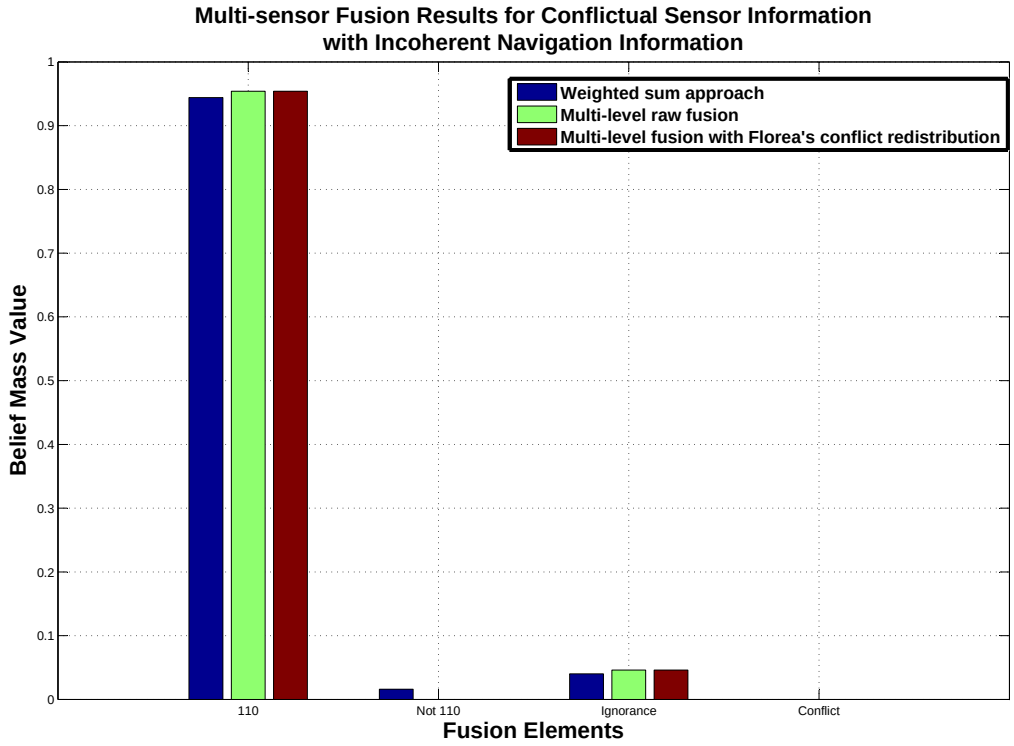



Figure 5.40: Multi-sensor Fusion Results for Conflictual Sensor Information with Incoherent Navigation Information

#### 5.5.6.4 Summary

In summary, the different simulation tests presented here have shown the improvements brought by the multi-criterion fusion for Speed Limit Assistance as it helps to determine the best navigation speed regarding to the criteria values, in other terms it helps to detect navigation errors. Then, it has also been shown that the enhanced multi-sensor fusion without conflict redistribution involves more often cases in which the final speed limit is *undefined*. This is due to the removal of the speed matching approach used by the previous Speed Limit Assistant. The latter always tries to find a match between the vision and the navigation speed to perform the multi-sensor fusion. There are consequently less cases in which the navigation speed matches the vision speed in the new approach. However, this behavior is safer as the generation of a conflict belief mass reveals a sensor false detection (under the assumption that the discernment frame is exhaustive), so reveals a case in which there is a high uncertainty over the effective speed limit. Finally, note that the multi-sensor approach also takes the quality of the sensor information into account contrary to the previous approach. This is possible through the consideration of a confidence variable value for the determination of the sensor belief masses.

### 5.5.7 Real-time Tests

Real-time tests have been carried out using the test car presented in Appendix.A equipped with the camera used for speed sign detection and the navigation system *ADASRP* providing the attributes for criteria evaluation. These tests have been obtained on roads providing a large number of speed signs while possessing the required attributes in the database. This

Table 5.11: Signification of the RTMaps HMI

| *Row Number* | *Information Type* | *Possible Characters* | *Signification* |
|---|---|---|---|
| $1^{st}$ | Navigation raw speed | All the speeds contained in the digital map | Speed extracted from the digital map |
| $2^{nd}$ | Navigation confidence variable | $\in [0, 1]$ | Quality of the navigation information |
| $3^{rd}$ | $MLCP$ value | 0<br>1<br>2 | $0.6 < 1 - \frac{MLCP}{MLCP_{max}} \leq 1.0$<br>$0.3 \leq 1 - \frac{MLCP}{MLCP_{max}} \leq 0.6$<br>$0.0 \leq 1 - \frac{MLCP}{MLCP_{max}} < 0.3$ |
| $4^{th}$ | ADAS Attribute status | 3<br>4 | Validated<br>Not validated |
| $5^{th}$ | Functional class status | 5<br>6 | Validated<br>Not validated |
| $6^{th}$ | Road Type | 7<br>8<br>9<br>10<br>11 | European<br>Highway<br>National<br>Departmental<br>Communal |
| $7^{th}$ | City status | 12<br>13 | In-city driving<br>Out-city driving |
| $8^{th}$ | Intersection Status | 14<br>15 | Intersection detected<br>No intersection detected |
| $9^{th}$ | Highway ramp Status | 16<br>17 | Highway ramp detected<br>No Highway ramp detected |
| $10^{th}$ | Guidance mode Status | 18<br>19 | Guidance mode activated<br>Guidance mode deactivated |
| $11^{th}$ | Final navigation speed | All the values of the discernment frame | Speed resulting from the multi-criterion fusion step |
| $12^{th}$ | Final navigation speed belief | $\in [0, 1]$ | Belief of the multi-criterion fusion selected speed |

section is dedicated to the presentation of the results obtained with the multi-level $SLA$ and organized similarly to the presentation of the simulation results: the multi-level fusion considering coherent navigation information and then the multi-level fusion considering incoherent navigation information. The latter corresponds to cases in which the multi-criterion fusion is the most efficient.

The results are here presented through figures corresponding to snapshots of the *RTMaps* Human-Machine Interface ($HMI$) components:

- The first block corresponds to the Vision information, so to the information obtained from the Speed Sign Recognition Algorithm. This information contains the previously and currently detected speeds in the first row, while the second row gives the confidence in the detection.

- The second block corresponds to the Navigation information necessary for the multi-criterion and multi-sensor fusion. This information is given using numerical identifiers whose signification is given in Table.5.11.

- The third block displays the results of the multi-sensor fusion. As mentioned in the previous sections, multi-sensor fusion results consist in two information:

  - The raw fusion results represented by the three first rows respectively corresponding to the final speed limit, its corresponding confidence and the value of the conflict generated during the multi-sensor fusion,
  - The fusion results obtained after conflict redistribution using *Florea*'s operator. This corresponds to the two last rows presenting the final speed limit after the redistribution and its belief value.

- A graphical display of the raw fusion results represented by an orange traffic sign placed in the middle of the figure regarding to the vision and navigation speeds represented by black and white traffic signs and respectively shown in the left and right parts.

- A snapshot of the video transmitted by the camera on which a graphical display of the conflict-redistributed fusion has been added. As for the raw fusion display, the conflict-redistributed fusion result is presented through a centered orange sign located between the black and white vision and navigation signs. Navigation data are also available at the bottom of this figure and correspond to the navigation information.

#### 5.5.7.1 Coherent Navigation Information

This section is composed of two cases which correspond to a configuration in which the sensors are concordant (so giving the same speed limit) and in which the sensors are in conflict (so giving different speed limit). These configurations are represented by Fig.5.41 and Fig.5.42. For each case, the navigation criteria are coherent with the real road context.

The first figure corresponds to a snapshot of the real-time tests obtained while driving on an European highway limited to $110km.h^{-1}$ which is well described by the navigation information. Indeed, the navigation speed is $110km.h^{-1}$ and the *Vehicle* is said to drive Out-city (13) on an European road (7) with a validated functional class (5). In addition, there is no intersection (15) and no highway ramp (17) with a guidance mode deactivated (19). Finally, the navigation confidence variable is ($C_{v_{nav}} = 0.68$) mainly due to the high value of the $MLCP$ (0) and the ADAS Attribute validation (3).

As the navigation speed ($110km.h^{-1}$) is coherent with the road context expressed by the different criteria, it is interesting to stress that this speed, which involves a set of three focal speeds ($50km.h^{-1}$ $70km.h^{-1}$ and $90km.h^{-1}$), is confirmed by the multi-criterion fusion. The latter represents the final navigation speed which is fused to the vision speed. On Fig.5.41, it can be seen that the navigation and the vision speeds are both equal to $110km.h^{-1}$, thus the raw multi-sensor results in a high confidence in this speed (0.95). In addition, as there is no conflict generation, the results of the multi-sensor fusion with conflict redistribution gives exactly the same results.

The second figure presents a snapshot of the Speed Limit Assistant on the same road configuration than for the previous figure. However, the difference lies in the detection of a new traffic sign by the $SLSR$: $50km.h^{-1}$. Sensors are consequently in conflict. On this figure, it can be seen that the confidence in the vision speed and in the navigation speed are high (respectively 0.96 and 0.81), the raw fusion is consequently subject to a high conflict of 0.69 which is too high for the raw fusion to give a final speed limit. The latter is then *undefined* contrary to the indicative speed limit of $110km.h^{-1}$ given after conflict redistribution. This speed reveals that the initial belief over the navigation information was stronger than the vision information belief. The conflict redistributed approach so results in a speed which corresponds to the real one ($110km.h^{-1}$). Finally, this conflictual situation originates from the detection of a speed limit

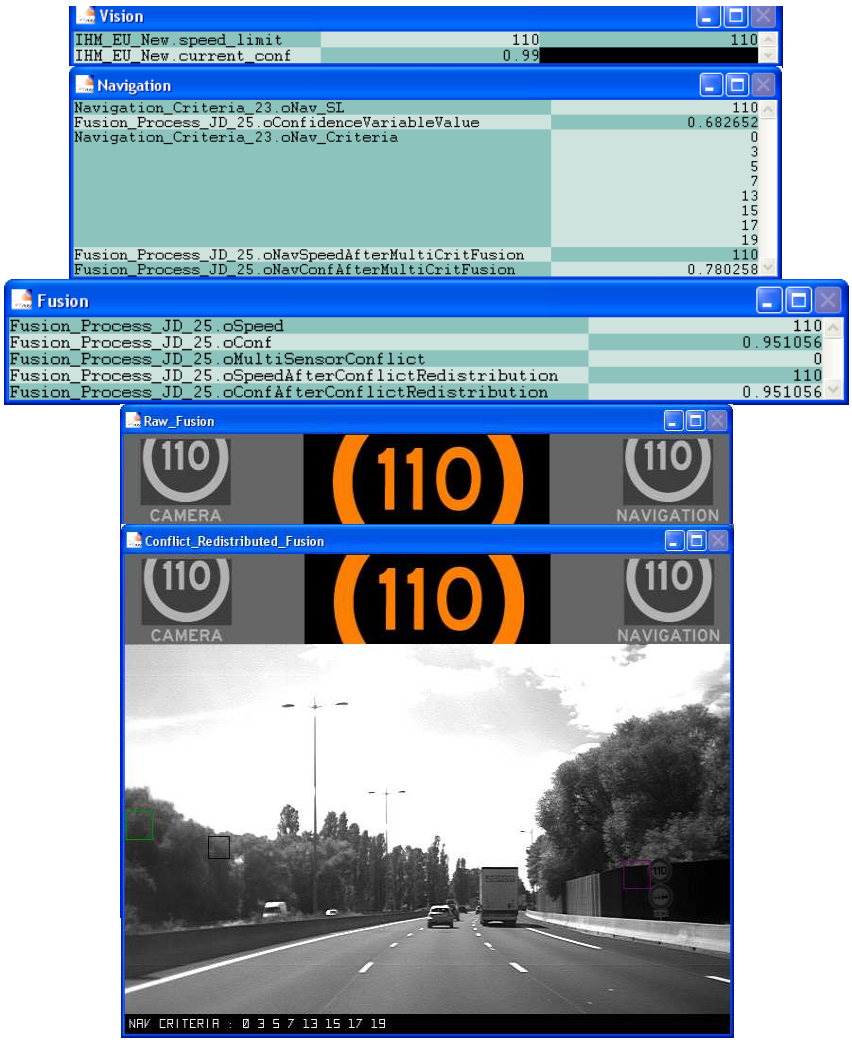



Figure 5.41: Real-time Fusion with Coherent Navigation Information and Concordant Sensors

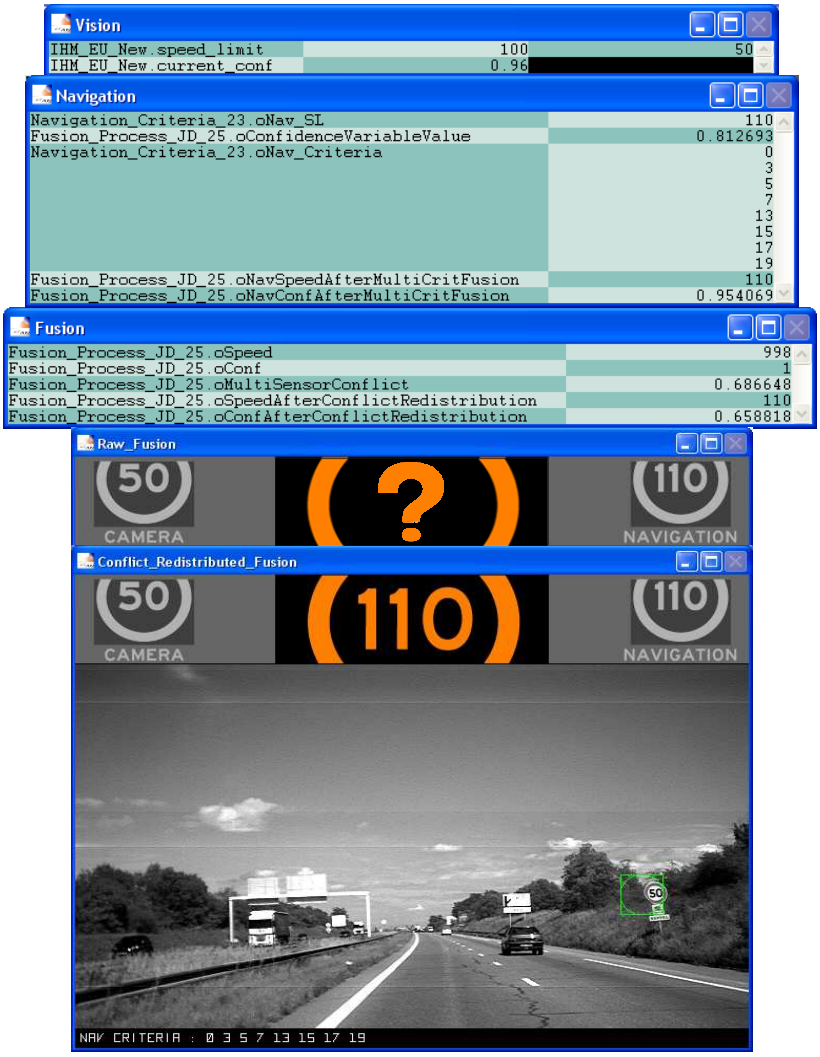



Figure 5.42: Real-time Fusion with Coherent Navigation Information and Discordant Sensors

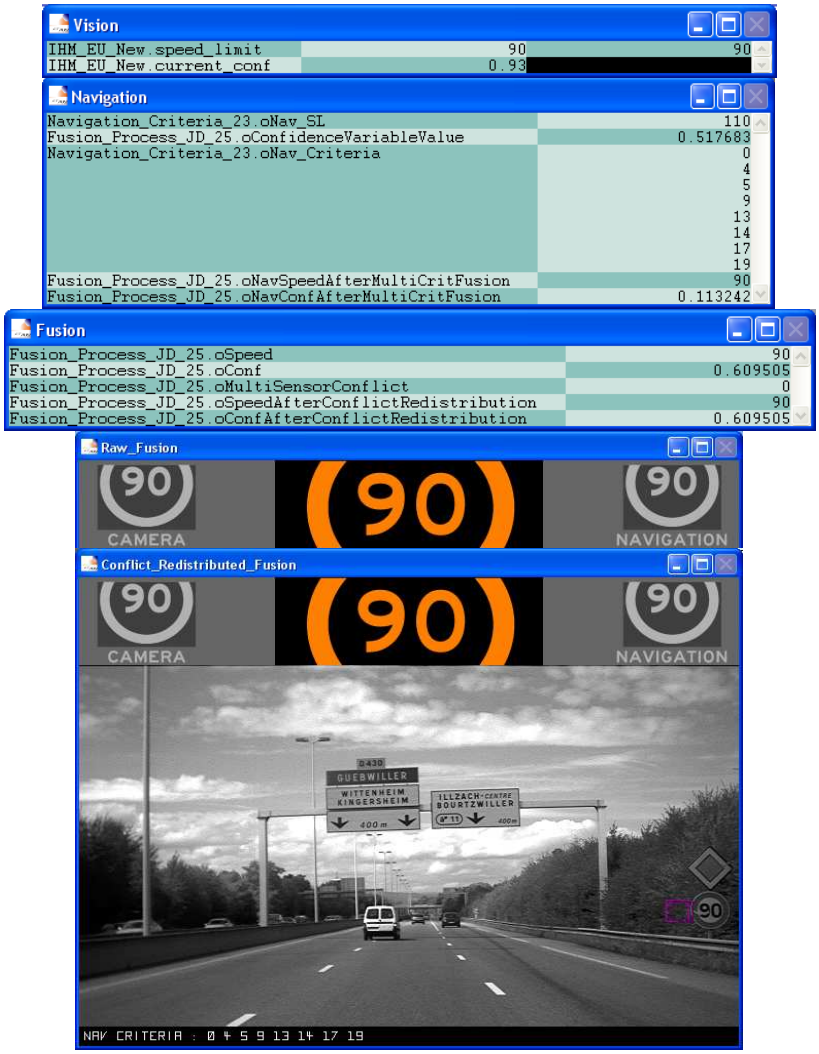



Figure 5.43: Real-time Fusion with Incoherent Navigation Information and Concordant Sensors

sign which is exclusively dedicated to trucks transporting dangerous materials. In other words, this situation is due to a false detection of the vision sensor.

#### 5.5.7.2 Incoherent Navigation Information

The first figure corresponds to a snapshot of the *SLA* obtained near an exit of a national $3_X3$ speedway which is limited to $90km.h^{-1}$. The navigation information is here slightly incoherent with this road context as the *Vehicle* is said to drive out-city (13) on a national road (9) which functional class is validated (5). If there are no highway ramp (17), the navigation has correctly detected the speedway exit (14). The guidance mode is still deactivated. Considering only these criteria, the navigation speed ($110km.h^{-1}$) seems not to be well adapted to the road context as French national roads are usually limited to $90km.h^{-1}$ . However, the most important information to be retained for this test is that this navigation speed of $110km.h^{-1}$ has been rejected by the multi-criterion fusion. Indeed, the final navigation speed is $90km.h^{-1}$. The multi-criterion fusion was consequently able to detect the navigation light incoherency and to

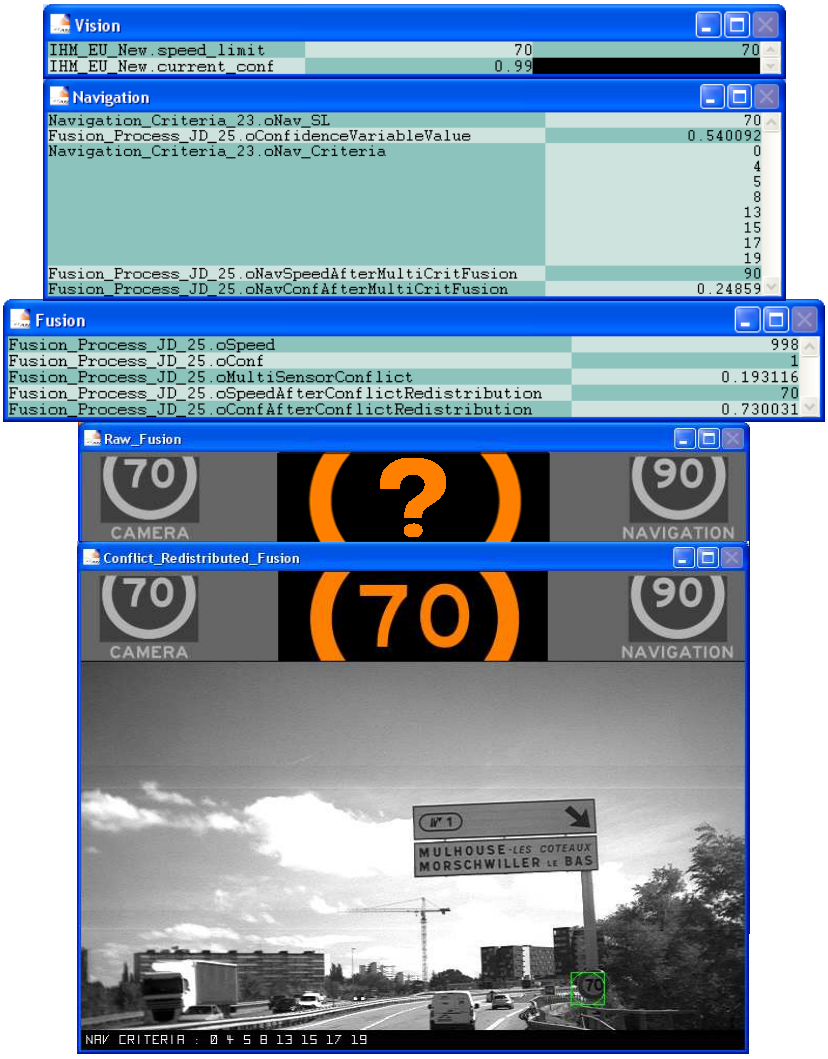



Figure 5.44: Real-time Fusion with Incoherent Navigation Information and Discordant Sensors

select the most suited speed regarding the road context. Furthermore, this fusion step was also able to integrate correctly the reliability of the navigation information. Indeed, the ADAS Attribute is not validated here (4), the digital map data may consequently be less accurate. This information has been considered in the calculation of the navigation confidence variable as $C_{v_{nav}} = 0.52$ is close to the threshold value of 0.5 (cf. Section.4.5.3). This results in the generation of a low belief in the navigation information as $m_n(90) = 0.11$.

Thanks to the multi-criterion fusion, the navigation inaccuracies and incoherency have been taken into account. The multi-sensor fusion is consequently performed over similar speeds, thus generates a high belief in $90km.h^{-1}$ (0.61). Then, as there is no conflict between sources, the conflict redistributed fusion gives the same results corresponding to the real speed limit of $90km.h^{-1}$.

The second figure (Fig.5.44) presents a snapshot of the Speed Limit Assistant obtained on a departmental $2_X2$ speedway limited to $90km.h^{-1}$. It can be noted that here, the navigation information is completely incoherent. In fact, the *Vehicle* is said to drive on an out-city (13) highway (8) which functional class is validated (5) but which is limited to $70km.h^{-1}$. Note

that no intersection (15) nor highway ramp (17) are detected and that the guidance mode is again deactivated. However, as for the previous test, the multi-criterion fusion has detected the incoherency and has selected a speed which is more related to the road context described by the criteria. Remind that this navigation speed involved the consideration of the set of focal speeds containing: $50km.h^{-1}$, $80km.h^{-1}$ and $90km.h^{-1}$. Here again, the navigation confidence variable is average (0.54) involving the generation of a low navigation belief (0.25).

Contrary to the previous test, the navigation selected speed is here in conflict with the vision speed ($70km.h^{-1}$). This results in the generation of conflict (0.19) and the selection of the *undefined* speed limit for the raw multi-sensor fusion. The conflict is low due to the small belief in the navigation information. Then, the conflict redistribution considers that the vision information is the most probable one as $70km.h^{-1}$ has been retained. Finally, the detected speed sign is dedicated to vehicles which are going to take the exit; as the *Vehicle* is still on the main road, this sign is not or is not yet valid. The vision information is consequently not adapted to the current driving situation and results in a false detection which has strong consequences on the $SLA$.

### 5.5.8 Summary

The benefits of the multi-level fusion has been proved in the previous sections through simulations and real-time tests. Indeed, the first fusion level, the multi-criterion fusion is of great help in the detection of false navigation information and in the integration of the positioning, localization and *Digital Map Database* reliability (i.e. in the integration of their inaccuracies). In addition, if it detects the navigation incoherences, it also selects the most suited speed regarding the road context described by the criteria. Consequently, this first level of fusion is very important for the multi-sensor fusion as it helps to avoid false $SLA$ results, thus shows the benefits of this PhD contribution over the $SLA$. In fact, previously, the navigation information was considered as inaccurate so that a match with the vision speed was always looked for. The proposed navigation improvements allowed to reconsider the weight that can be imputed to this information at a level similar to the vision information. Furthermore, the different tests carried out in real-time have shown that most of the false $SLA$ results are due to the vision information. Indeed, the speed sign recognition is not deterministic and is subject to false detections. These false detections can be, as presented in Fig.5.45, related to a recognition problem (cf. (a) and (b)), to the consideration of a sign which is not dedicated to the used *Vehicle* (cf. (c)), to the consideration of numbers which do not refer to a speed limit (cf. (d)). Future studies will consequently be firstly focused on the improvement of the $SLSR$.

Nevertheless, if the proposed Speed Limit Assistant presents interesting results, some improvements can still be done. It has been shown in the third real-time test (cf. 5.43), that the multi-criterion fusion has selected the most suited speed regarding the criteria and the available focal speeds. However, regarding these criteria, a higher speed such as $110km.h^{-1}$ may have been more suited to the road context, thus reveals the necessity to refine the focal speed table. Then, the addition of a new sensor may also be of great interest to enhance the quality of the Speed Limit Assistant and as it would help to exploit more efficiently the benefits brought by the *Florea*'s redistribution operator[30]. Another research line may also lie in the consideration of a multi-hypothesis tracking, thus implies the consideration of chronological information into the global fusion scheme.

[30]Remind that this operator only redistributes the conflict over the sources which generates it.

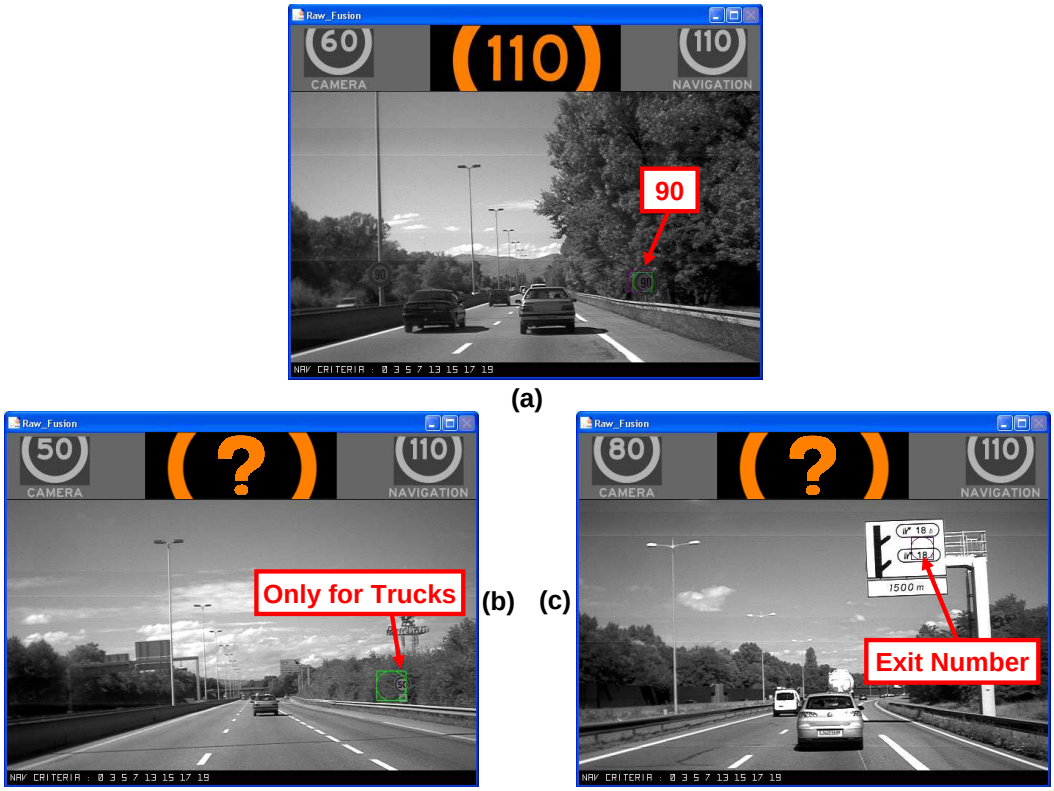



Figure 5.45: Speed Sign Recognition Limitations

## 5.6 Conclusion

In this Chapter the results obtained by applying this PhD contributions to Navigation-aided *ADAS* have been presented in three steps: the unconstrained trajectory generation results, the constrained trajectory generation results and the *SLA* results. From this Chapter, two main elements have been highlighted:

- The application of the unconstrained trajectory generation to the Longitudinal control of the test *Vehicle* has shown that the constraints related to the *Driver*, the *Vehicle* and its *Environment* have been considered in the Longitudinal Controller synthesis step. For instance, the latter has to modify the limit speed profile calculated from the trajectory in order to perform comfortable braking phases. Contrary to this, the results obtained with constrained trajectories applied on the Lateral Control of the test *Vehicle*, and especially the trajectory minimizing its strain energy ($WEM$ trajectory), has presented interesting results. Indeed, if all the constrained trajectories are contained in the validity area (so grants part of the geometrical constraints), the $WEM$ trajectory uses more efficiently the lateral space available in this area. This results in smoother trajectories, i.e. trajectories characterized by a smoother curvature. This benefits seems to be confirmed by the results obtained through the application of the $WEM$ trajectory to the Lateral Controller. Indeed, considering the same test conditions, the $WEM$ trajectory seems to be less power consuming and to provide better precision than the other constrained trajectory and than the unconstrained trajectory. These comparisons are consequently showing the benefits of the constraints management which consists in the dispatching of the results over the

*Controller* **and** the *Reference*.

- On the other hand, the application of the multi-level data fusion to the $SLA$ has shown several benefits compared to the conventional approach presented in [Bradai, 2007]. Indeed, the considered modeling and estimation strategies, based on the application of the *Rombaut/Gruyer* model now takes all the navigation inaccuracies into account, thus characterizing the reliability of the sensor information more efficiently. Moreover the multi-criterion fusion, is more suited to the detection of the navigation incoherences and to the selection of the best navigation speed regarding the road context (described by the criteria). Then, the new decision approach consisting in providing two information to the *Driver*, is more safe. Indeed, the raw fusion does not automatically redistribute the conflict over the different propositions. This allow the $SLA$ to give the *undefined* speed as a results of the fusion in high conflict case. Finally, as this raw fusion involves several cases in which the $SLA$ is undecided, the *Florea*'s conflict redistribution operator has been used to provide an indicative speed to the *Driver*. Anyway, these results have shows the necessity of information combination for Navigation-aided $ADAS$. Indeed, regarding to the considered sensors (the navigation system and the camera coupled to a $SLSR$), a great amount of information has to be processed to provide only relevant information to the *Driver*. In addition, it has also been shown that the sensor information may be erroneous, redundant, inaccurate, etc., thus enforcing the information combination necessity.

# General Conclusion

During the last decades, due to several alarming study results, road safety has become one of the most dynamic automotive research field. Indeed, the constant increase of road traffic involves the driving task to be more difficult, thus generating more situations in which the *Driver* process capabilities are exceeded. Assisting the *Driver* in his driving task consequently constitutes a clue to road fatalities reduction. Considering this statement, several systems have been developed to perform specific assistance tasks such as Cruise Control. However, to significantly reduce road injuries a new generation of assistance systems named Advanced Driver Assistance Systems (*ADAS*) have to be designed. The development of these systems constitutes the core subject of the present PhD as mentioned in Chapter.2. Indeed, contrary to the previous generation of assistance systems, *ADAS* will need to consider more aspects of the *Vehicle* and its *Environment*. Consequently, as they aim at assisting the *Driver* in his driving task, they have to provide relevant information to the *Driver* (passive assistance) and/or to perform safe and comfortable control of the *Vehicle* actuators (active assistance). To perform these tasks, this PhD has proposed two main contributions:

- It has been shown in Chapter.2 that providing efficient assistance to the *Driver*, requires *ADAS* to evaluate the *Vehicle* global situation, i.e. requires information about the *Vehicle* status and the composition of the *Environment*. Usually, in the automotive domain, information is provided by sensors. The situation evaluation is then obtained through the combination of this information. This is a challenging task as sensor information may be redundant, inaccurate, etc. One of this PhD contribution consists in an information management approach based on a specific data fusion theory. The latter, based on the Dempster-Shafer Theory, consists in a two-level fusion structure dedicated to speed limit determination regarding a navigation and a vision system. The benefits of this solution are multiple and have been shown through simulations and real-time tests in Chapter.5. On the one hand, the first level of fusion dedicated to the determination of the sensor information presents the great advantage to consider the sensors reliability directly during the sensor belief mass estimation. This is mainly due to the consideration of the *Rombaut/Gruyer* model and to the determination of confidences variables which are especially dedicated to the evaluation of the sensors reliability. These variables are also of great help for the multi-criterion fusion. Indeed, the latter, used to determine the navigation information and based on criteria describing the road context, helps to detect the navigation errors and to select the speed which best suits the considered road context. Finally the consideration of the conflict as an additional source of information after the multi-sensor combination, has revealed to be an interesting solution for Speed Limit Determination. In fact, in high conflict situation, the Speed Limit Assistant stays undecided about the final speed limit. This represents a safe decision compared to the automatic conflict redistribution usually processed in conventional approaches.

- Besides the evaluation of the current driving situation, *ADAS* have to consider the constraints related to the *Driver*, the *Vehicle* and the *Environment* to avoid the generation of hazardous situation which may lead to accidents. Usually, in the automotive domain, these constraints are considered during the *Controller* synthesis step. This approach has been considered in this PhD through the application of the proposed unconstrained trajectory generation to the Longitudinal Control of the *Vehicle*. Indeed, only *a priori* information about several points location have been used by a Parametric Cubic Spline model to generate those trajectories. The constraints related to the *Driver* comfort or to the *Vehicle* acceleration/deceleration capabilities were then considered during the synthesis of the Longitudinal Controller. Even if the proposed Longitudinal Controller has presented interesting results as mentioned in Chapter.5, these results and the description of the techniques used to integrate these constraints in the *Controller*, have also shown the high complexity of this synthesis step.

  To cope with this problem, a constrained trajectory generation approach has been presented. As mentioned by its name, this approach is based on the consideration of the constraints related to the *Driver*, the *Vehicle* and the *Environment* directly in the generation of the trajectories. This has been done by considering the trajectory generation from a different point of view. Contrary to the unconstrained approach which forced trajectories to interpolate *a priori* information, the constrained approach defines an area in which the trajectory is allowed to lie. In this area, there are an infinity of possible trajectories. To select the best trajectory in the set of possible ones, the trajectory generation has been considered as an optimization problem. This helps to define the optimal trajectory regarding a cost criterion. For comparison purpose different cost criteria have been considered as mentioned in Chapter.3. The results obtained by applying these trajectories to the Lateral Control of the test *Vehicle* have been presented in Chapter.5 and have highlighted two major elements. First, constrained trajectories, as they are already integrating the constraints related to the *Driver*, the *Vehicle* and the *Environment*, allows the Lateral Controller to consider additional constraints which are not yet considered in the *Reference* generation. Then the constrained trajectory which minimizes its strain energy has been shown to be very interesting. Indeed, the latter uses more efficiently the space available in the area, thus provide smoother trajectories than the other cost criterion minimizations. Moreover, it seems that this trajectory involve a reduction of the energy required by the *Vehicle* to perform its tracking.

If the application of this PhD contributions over *ADAS* are presenting satisfactory results, a lot of improvements can still be done at different levels.

1. From the *ADAS* point of view, several improvements can be done to the Speed Limit Assistant and to the constrained trajectory generation:

   - The constrained trajectory which minimizes its strain energy has been shown to be the most interesting one. Indeed, as it takes several constraints into account, it is well suited to automotive applications such as trajectory tracking (i.e. Lateral Controller). Nevertheless, it has been defined regarding assumptions which may limit its potential benefits. For instance, the non-linear solution set of the inequalities defining the polynomial positivity (cf. Section3.5.2.2), could be considered by the optimization algorithm. Then, another perspective could be the reduction of the assumptions used for the strain energy estimation (simultaneous minimization of each parametric

curvatures $\kappa_x$ and $\kappa_y$ as well as considering that $\dot{x}(t)^2$ and $\dot{y}(t)^2$ are small compared to 1).

- Tests done upon the Speed Limit Assistant have shown that, now, the inaccuracies and the errors of the navigation information are well considered and that the major source of errors can be imputed to the vision system. The first research line for this $ADAS$ could then be focused on the improvement of the $SLSR$ to avoid these false detections. Another improvement could be to refine or to remove the navigation focal element table to give the best navigation speed regarding the context and the complete discernment frame. Finally, more *Digital Map Database* attributes describing the road context, could be taken into account during the multi-criterion fusion. This would help to enhance the evaluation of the navigation reliability.

2. From a higher level of abstraction, several enhancements to the employed trajectory generation and multi-level data fusion strategies are available:

   - The constrained trajectory generation is based on the consideration of constraints related to the *Driver*, the *Vehicle* and the *Environment*. A first research line could then consist in the addition of constraints related to these elements. For instance this could refer to the consideration of a maximum jerk value for *Driver* comfort, a maximum *Vehicle* velocity, etc.; this opens the door to the consideration of different or mixed cost criteria. Finally, another research line could be the integration of the third dimension to generate 3-dimensional constrained trajectories. With the road bank and slope, the possibility to consider and to minimize the global *Vehicle* energy consumption could then be allowed.
   - If the multi-level and the multi-criterion fusion are of great help in Speed Limit Determination, the employed techniques and especially the sensor reliability evaluation benefits could be even more validated by considering additional sensors, or by the information coming from another *Vehicle*. Finally, the current discernment frame could be enhanced through a transposition into the *open extended world*. Indeed, as usually in real applications, all the solutions of a given problem could not be explicitly defined.

# Appendix A

# Test Vehicle

## A.1 Introduction

The contributions presented in this PhD have been implemented in the $MIPS - MIAM$ test car presented in Fig.A.1. The latter corresponds to a classical *Renault Scenic* car model in which the different elements required for the test of navigation-based developments have been added. Indeed, it contains several sensors and actuators necessary for a few control-oriented $ADAS$ applications. In addition this test car includes the softwares which aim to select, manage and generate information required for the considered $ADAS$ as presented in Fig.A.2.

## A.2 Hardware

The hardware components can be classified into three categories: the sensors, the management devices, and the actuators.

### A.2.1 Sensors

The main sensors available in the test car are:

- The speed sensor: already embedded in the *Vehicle* gearbox, its signal is transmitted to the *Vehicle* calculator and finally sent to the data acquisition unit (*Autobox*). The acquired signal is a frequency variable square signal. A simple frequency measurement is consequently giving the current *Vehicle* speed. This represents an important information for the Longitudinal Controller as its comparison to the speed reference determines the control signals to be sent to the actuators.

- A *SensorBox*: designed by *NAVTEQ*, it is composed of a *GPS* receiver, gyroscopes, and a temperature sensor. This information is combined to several information related to the *Vehicle* (speed, direction, etc.) to generate the data transmitted to *ADASRP*. This *SensorBox* is used for the positioning and the map-matching of the *Vehicle* on the *Digital Map Database* and is therefore required for the Longitudinal controller. The maximum acquisition rate of the *SensorBox* is $10Hz$.

- A *DGPS*: the *GPS* sensor available in the *SensorBox* is not accurate enough for the Lateral Controller. Indeed, the *SensorBox* gives classical accuracies of $5m$ in absolute and $2m$ in relative (cf. Section.2.3.3) while the Lateral Controller requires a higher accuracy. To cope

Figure A.1: Used Test Car

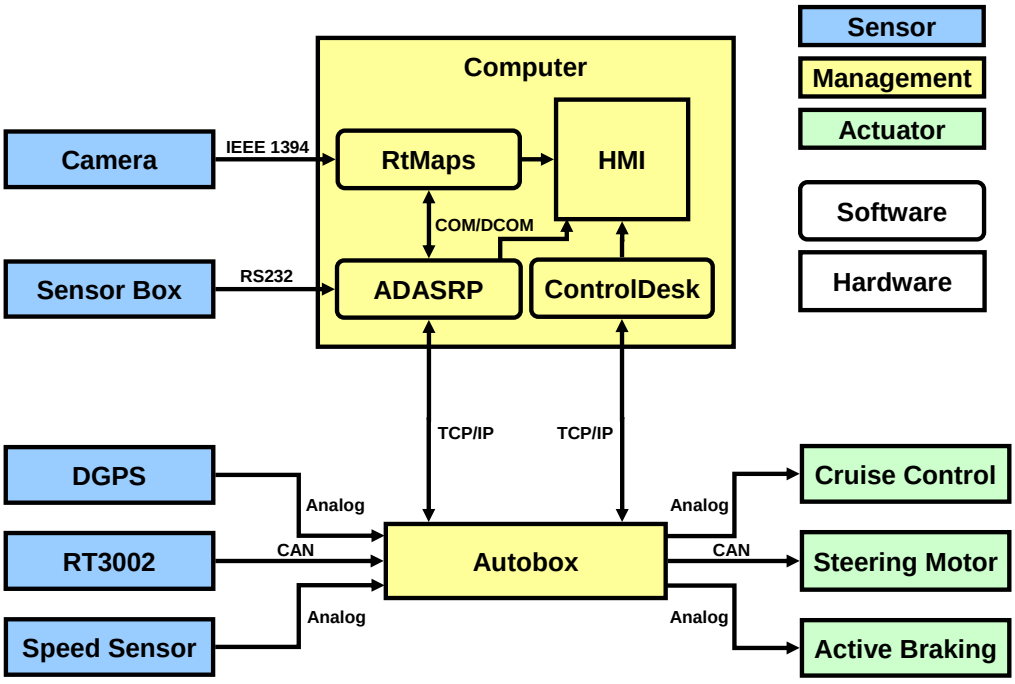



Figure A.2: Test Car Functional Scheme

with this problem, a *DGPS* sensor, composed of a reference station (*Scorpio SK6002*) and of a mobile station (*Aquarius MK5002*), has been used for the Lateral Controller validation. The reference station is fixed and sends correction signals to the mobile station through an $UHF$ radio signal communication. The acquisition rate of the *DGPS* sensor is $10Hz$.

- A *RT3002*: provided by *OXTS*®, it is a well-known inertial sensor which includes three angular rate sensors (gyrometers), three servo-grade accelerometers and a *GPS* receiver. In addition, it contains the different algorithms dedicated to the process of the different data. This sensor is used by the Longitudinal Controller to check the *Vehicle* accelerations and by the *Lateral Controller* as it provides the different *Vehicle* data (speed, angles, accelerations, etc.).

- The Camera used for the Speed Limit Assistant is a *ECK-101* provided by *Sensata Technologies*. It is specially designed for vehicle-embedded applications because of its small size and as it adapts in real-time the contrast using the *AutoBrite* technology. Moreover it provides 30 non-colored frame per second with a 640 by 480 resolution.

### A.2.2 Management

The management components correspond to devices which interface the different *Vehicle* components. These devices are consequently used by all the *ADAS* applications of the test car:

- An *Autobox* from *DSpace*®: it is the data acquisition device which acts as a link between the sensors, the computer and the actuators of the test *Vehicle*. This device is well known and widely used in the automotive domain as it is particularly robust and efficient. Indeed, it provides input/outputs boards which can transmit signals of different nature: Serial, CAN bus, Analog, Digital, etc. In addition, the *Autobox* can be linked to a computer via the TCP-IP protocol and thus is able to exchange information with the computer in real-time. Finally, it presents a great advantage for *Matlab/Simulink*-based controllers as these models can be compiled, loaded and directly executed from the *Autobox*.

- An industrial computer: it is used for supervision through a Human Machine Interface (*HMI*) but also for data processing as it contains the different softwares and especially *ADASRP*. This computer runs with a $2.13GHz$ dual core CPU and $4GB$ RAM.

### A.2.3 Actuators

The main actuators available in the test car are:

- The Cruise Control ($CC$): it acts directly on the throttle angle regarding to the different commands given by the *Driver* via a remote. However, to automatically control the *Vehicle* on road, this remote has been replaced by signals coming from the *Longitudinal Controller*. The internal $CC$ control loop (aiming at cruising when active) has not been modified. The proposed speed assistant only adapts the cruising reference. Control signals are:

  - *Accelerate* ($ACC$) which increases the throttle angle. A short command on this signal ($< 1s$) increases the throttle angle by a single step ($2-3km/h$) and a long command ($> 1s$) increases the speed by gradual steps until the end of the command. The final speed is then memorized and maintained by the Cruise Control system.

Figure A.3: The Steering Motor

- *Decelerate* ($DC$) which decreases the throttle angle. A short command on this signal ($< 1s$) slightly decreases the *Vehicle* speed ($2-3km/h$) while a long command ($> 1s$) gradually decreases the *Vehicle* speed. The final speed is then saved and maintained.
- *Cancel* which cancels a maintain phase or another $ACC$ or $DC$ command. The *Vehicle* is then put into idle mode.

This actuator is used by the Longitudinal Controller.

- The Active Braking system: it is composed of a brake pedal and a vacuum booster which can be electrically controlled. However, the *Driver* can still use the braking pedal himself. This actuator is, as the Cruise Control, used to manage the *Vehicle* speed, so used by the Longitudinal Controller.

- The steering motor: exclusively used by the Lateral Controller, this motor is a multipolar permanent magnets synchronous brushless, three-phases motor *SKADDR 148-90* from *Motor Power Company*® which is directly mounted on the steering column. The motor provides a stall torque (the torque which is produced by the motor when the output rotational speed is zero) of $20Nm$ with a maximum rotational speed of $150rad/min$ and is driven by its own servo drive. The performances of this motor are certainly oversized for the considered application and the dimensions of the solution may seem important. However, the specifications that must be reached requires this solution. The motor has been sized to steer the wheel when the *Vehicle* is stopped. The speed performances are required for *Vehicle* identification process when sinusoidal input at high frequency ($\geq 5Hz$) are done. Finally, this motor, presented in Fig.A.3, has no mechanical transmission (no chain and no belt) which is better for security purposes. The steering motor is connected to the CAN bus of the *Vehicle*.

## A.3 Software

As presented in the previous section, the different sensors and actuators are managed by the *Autobox* and the industrial computer. The latter contains the softwares which are used for the development of new *ADAS*. Among these softwares, the three most important ones are detailed here: *ADASRP*, *Matlab/Simulink* and *RTMaps*.

- *Advanced Driver Assistance Systems Research Platform* (*ADASRP*). This software is part of the navigation system developed and provided by *NAVTEQ*®. It is specially dedicated to the development of new *ADAS*. Indeed, it provides several useful software elements such as a *Map-Matching algorithm*, an *Electronic Horizon Provider*, a *Digital Map Database* and a graphical interface (cf. Section.2.2.2.2). Moreover, it is based on an architecture allowing quick access to digital map data, so to their extraction for specific purposes. Furthermore, it can be combined to the sensors related to the navigation system presented in the previous section. This software is coded in *C++* under *Visual Studio 2005*. It contains the different developments related to the trajectory generation used for the Longitudinal and Lateral controllers.

- *Matlab/Simulink*: it is a well known software developed by *MathWorks*® mainly used for educational, research and development purposes. Indeed, it represents an environment and an accessible programming language which can be used for various applications. This software is here used to design controllers. Furthermore, using a specific communication protocol (*C-library* technology developed by *Dspace*), the exchange of information between the loaded model and *ADASRP* is possible.

- *Real-Time Multisensor Advanced Prototyping Software* (*RTMaps*): developed by *Intempora*, it is specially designed for signal processing purpose. Indeed it provides a graphical-based programming which allows real-time multiple-data acquisitions, data fusion and processing, at high flow rate. In addition, each sensor data is precisely dated, so giving a common time basis for all the considered signals. This software is here used for the integration of the data fusion elements related to the Speed Limit Assistant.

- *ControlDesk*: developed by *DSpace*, *ControlDesk* is a human-machine interface used to monitor data during real-time tests and specially data coming from other *DSpace* devices. It also provides the possibility to modify in real-time some parameters of the compiled *MatLab/Simulink* model loaded in the *Autobox*.

# Appendix B

# Discrete Suboptimal Energy Criterion

The purpose of this section is to give some details about the transformation of the suboptimal energy continuous expression (B.1) into its discrete form presented in (B.2).

$$\tilde{E} = \int_{t_0}^{t_n} \left(\ddot{x}^2 + \ddot{y}^2\right) dt \tag{B.1}$$

$$\tilde{E} = \sum_{i=0}^{n-2} \frac{4h_i}{3} \left(b_{f_{x_i}}^2 + b_{f_{x_i}} b_{f_{x_{i+1}}} + b_{f_{x_{i+1}}}^2\right) + \sum_{i=0}^{n-2} \frac{4h_i}{3} \left(b_{f_{y_i}}^2 + b_{f_{y_i}} b_{f_{y_{i+1}}} + b_{f_{y_{i+1}}}^2\right) \tag{B.2}$$

To do this let consider the general $PCS$ expression defined by (B.3) and the example presented in Fig.B.1 top plot.

$$f_i(t) \begin{cases} f_{x_i}(t) = a_{f_{x_i}} t^3 + b_{f_{x_i}} t^2 + c_{f_{x_i}} t + d_{f_{x_i}} \quad t \in [t_i, t_{i+1}] \\ f_{y_i}(t) = a_{f_{y_i}} t^3 + b_{f_{y_i}} t^2 + c_{f_{y_i}} t + d_{f_{y_i}} \quad t \in [t_i, t_{i+1}] \end{cases} \tag{B.3}$$

The $PCS$ second derivative is then defined by the following linear expressions:

$$\ddot{f}_i(t) \begin{cases} \ddot{f}_{x_i}(t) = 6a_{f_{x_i}} t + 2b_{f_{x_i}} \text{ for } t \in [t_i, t_{i+1}] \\ \ddot{f}_{y_i}(t) = 6a_{f_{y_i}} t + 2b_{f_{y_i}} \text{ for } t \in [t_i, t_{i+1}] \end{cases} \tag{B.4}$$

Fig.B.1, which focuses on the $x$ coordinate, shows on the bottom plot that, for each interval $[t_i, t_{i+1}]$ the evolution of the second derivative is defined by a linear variation going from $2b_{f_{x_i}}$ to $2b_{f_{x_{i+1}}}$. The linear expressions can consequently be defined regarding these Spline coefficients such that:

$$\ddot{f}_i(t) \begin{cases} \ddot{f}_{x_i}(t) = 2\left(\frac{b_{f_{x_{i+1}}} - b_{f_{x_i}}}{h_i} t + b_{f_{x_i}}\right) \text{ for } t \in [t_i, t_{i+1}] \\ \ddot{f}_{y_i}(t) = 2\left(\frac{b_{f_{y_{i+1}}} - b_{f_{y_i}}}{h_i} t + b_{f_{y_i}}\right) \text{ for } t \in [t_i, t_{i+1}] \end{cases} \tag{B.5}$$

with $h_i = t_{i+1} - t_i$.
The integration of (B.5) into (B.1) gives:

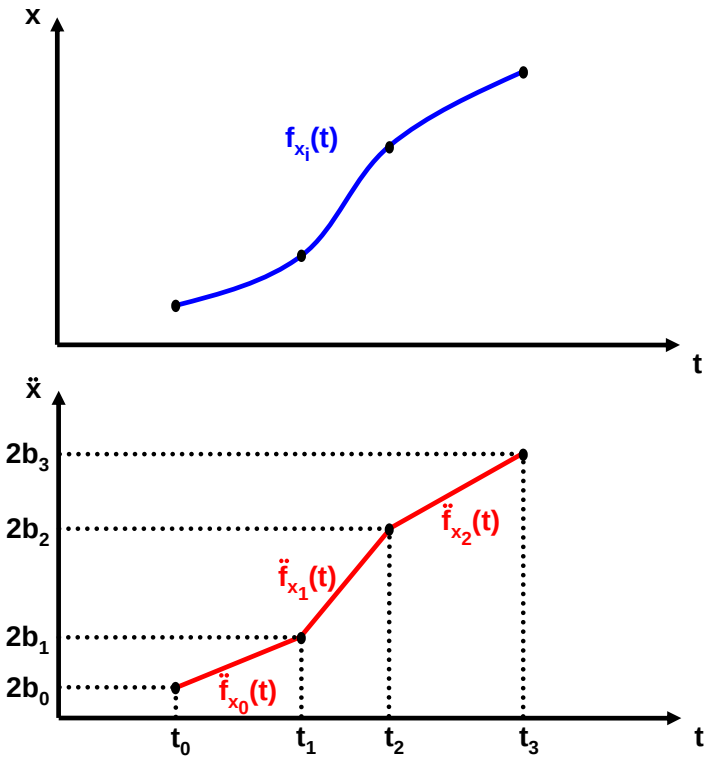



Figure B.1: Spline Second Derivative Representation

$$
\begin{aligned}
\tilde{E} = & \sum_{i=0}^{n-2} \int_{t_i}^{t_{i+1}} \left( \left( 2 \left( \frac{b_{f_{x_{i+1}}} - b_{f_{x_i}}}{h_i} t + b_{f_{x_i}} \right) \right)^2 + \left( 2 \left( \frac{b_{f_{y_{i+1}}} - b_{f_{y_i}}}{h_i} t + b_{f_{y_i}} \right) \right)^2 \right) dt \\
\tilde{E} = & \sum_{i=0}^{n-2} 4 \int_{t_i}^{t_{i+1}} \left( \frac{b_{f_{x_{i+1}}} - b_{f_{x_i}}}{h_i} t + b_{f_{x_i}} \right)^2 dt + \sum_{i=0}^{n-2} 4 \int_{t_i}^{t_{i+1}} \left( \frac{b_{f_{y_{i+1}}} - b_{f_{y_i}}}{h_i} t + b_{f_{y_i}} \right)^2 dt \\
\tilde{E} = & \sum_{i=0}^{n-2} 4 \int_{t_i}^{t_{i+1}} \left( \frac{t^2}{h_i^2} \left( b_{f_{x_i}}^2 - 2 b_{f_{x_i}} b_{f_{x_{i+1}}} + b_{f_{x_{i+1}}}^2 \right) + \frac{t}{h_i} \left( 2 b_{f_{x_i}} b_{f_{x_{i+1}}} - 2 b_{f_{x_i}}^2 \right) + b_{f_{x_i}}^2 \right) dt \\
& + \sum_{i=0}^{n-2} 4 \int_{t_i}^{t_{i+1}} \left( \frac{t^2}{h_i^2} \left( b_{f_{y_i}}^2 - 2 b_{f_{y_i}} b_{f_{y_{i+1}}} + b_{f_{y_{i+1}}}^2 \right) + \frac{t}{h_i} \left( 2 b_{f_{y_i}} b_{f_{y_{i+1}}} - 2 b_{f_{y_i}}^2 \right) + b_{f_{y_i}}^2 \right) dt \\
\tilde{E} = & \sum_{i=0}^{n-2} 4 \left( \frac{h_i}{3} \left( b_{f_{x_i}}^2 - 2 b_{f_{x_i}} b_{f_{x_{i+1}}} + b_{f_{x_{i+1}}}^2 \right) + \frac{h_i}{2} \left( -2 b_{f_{x_i}}^2 + 2 b_{f_{x_i}} b_{f_{x_{i+1}}} \right) + h_i b_{f_{x_i}}^2 \right) \\
& + \sum_{i=0}^{n-2} 4 \left( \frac{h_i}{3} \left( b_{f_{y_i}}^2 - 2 b_{f_{y_i}} b_{f_{y_{i+1}}} + b_{f_{y_{i+1}}}^2 \right) + \frac{h_i}{2} \left( -2 b_{f_{y_i}}^2 + 2 b_{f_{y_i}} b_{f_{y_{i+1}}} \right) + h_i b_{f_{y_i}}^2 \right) \\
\tilde{E} = & \sum_{i=0}^{n-2} \frac{4 h_i}{3} \left( b_{f_{x_i}}^2 + b_{f_{x_i}} b_{f_{x_{i+1}}} + b_{f_{x_{i+1}}}^2 \right) + \sum_{i=0}^{n-2} \frac{4 h_i}{3} \left( b_{f_{y_i}}^2 + b_{f_{y_i}} b_{f_{y_{i+1}}} + b_{f_{y_{i+1}}}^2 \right)
\end{aligned}
\tag{B.6}
$$

# Appendix C

# Lateral Controller Synthesis

## C.1 Presentation of the MPC Control Solution

To follow a path $\mathcal{P}$, the main physical quantity which is taken into account is the lateral displacement of the *Vehicle*, commonly called $y(t)$. The aim of the $MPC$ solution is to find the optimal control signal $\Theta_w$ which helps to follow the points $y_{ref}(i)$ of the path $\mathcal{P}$. Then, the cost criterion is defined by:

$$J(k) = \sum_{n=1}^{N_p} |\hat{y}(k+n) - y_{ref}(k+n)|_Q + \sum_{n=0}^{N_c} |\Theta_w(k+n)|_R \tag{C.1}$$

The prediction of $y$ on the prediction horizon is based on *Vehicle* dynamic signals and can be expressed as:

$$\dot{y}(t) = V_y(t)cos(\psi(t)) + V_x(t)sin(\psi(t)) \tag{C.2}$$

such that $\psi(t)$ is the heading angle of the *Vehicle* defined by:

$$\dot{\psi}(t) = r(t) \tag{C.3}$$

At each sample time, the path is projected in a local referential whose origin is the *Vehicle* $CoG$, the axis $\vec{x}$ coincides with the axis of the *Vehicle* and the axis $\vec{y}$ is defined laterally from the *Vehicle*. Fig.C.1 presents the path in the ground fixed axes and the new coordinates ($\vec{x}$, $\vec{y}$).

Considering the hypothesis that the heading angle $\psi(t)$ stays relatively small on the prediction horizon, (C.2) is linearized considering usual trigonometric function simplifications:

$$\dot{y}(t) \approx V_y(t) + V_x(t)\psi(t) \tag{C.4}$$

As it can be seen in (C.3) and (C.4), the *Vehicle* dynamic signals $r(t)$ and $V_y(t)$ are used to obtain the lateral displacement of the *Vehicle*. Finally, the prediction model is a $LPV$ model with 1 input ($\Theta_w$), 1 output ($y$), 4 states ($V_y$, $r$ , $y$ and $\psi$) and 1 varying parameter ($V_x$) such that:

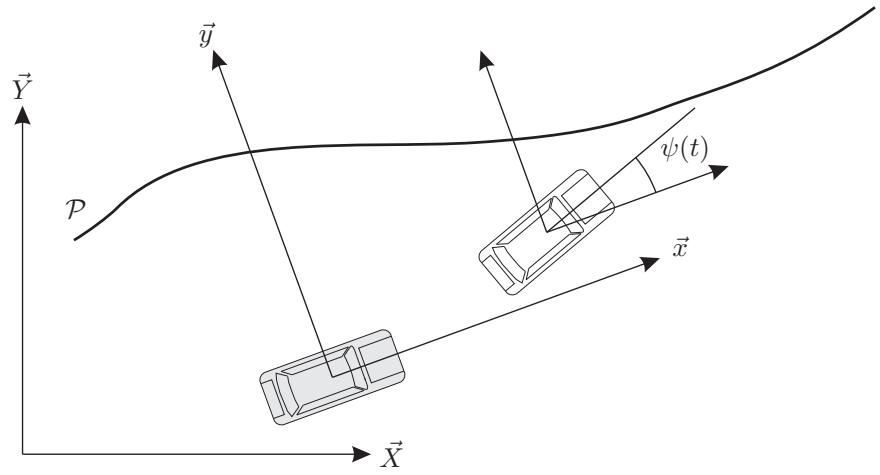



Figure C.1: Adaptating the Referential

$$\begin{Bmatrix} \dot{V}_y \\ \dot{r} \\ \dot{y} \\ \dot{\psi} \end{Bmatrix} = \begin{bmatrix} \frac{-CS_f(V_x)-CS_r(V_x)}{MV_x} & \frac{-l_fCS_f(V_x)+l_rCS_r(V_x)}{MV_x} - V_x & 0 & 0 \\ \frac{-l_fCS_f(V_x)+l_rCS_r(V_x)}{J_v} & \frac{-l_f^2CS_f(V_x)-l_r^2CS_r(V_x)}{J_vV_x} - V_x & 0 & 0 \\ 1 & 0 & 0 & V_x \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{Bmatrix} V_y \\ r \\ y \\ \psi \end{Bmatrix} + \begin{bmatrix} \frac{CS_f(V_x)}{M} \\ \frac{l_fCS_f(V_x)}{J_v} \\ 0 \\ 0 \end{bmatrix} \theta_w \tag{C.5}$$

This model is sampled with a simple first order approximation of the derivative such that:

$$\begin{Bmatrix} V_y(k+1) \\ r(k+1) \\ y(k+1) \\ \psi(k+1) \end{Bmatrix} = A_k \begin{Bmatrix} V_y(k) \\ r(k) \\ y(k) \\ \psi(k) \end{Bmatrix} + B_k\theta_w(k) \tag{C.6}$$

with

$$A_k = \begin{bmatrix} \frac{-CS_f(V_x)-CS_r(V_x)}{MV_x}T_e + 1 & \left(\frac{-l_fCS_f(V_x)+l_rCS_r(V_x)}{MV_x} - V_x\right)T_e & 0 & 0 \\ \frac{-l_fCS_f(V_x)+l_rCS_r(V_x)}{J_v}T_e & \left(\frac{-l_f^2CS_f(V_x)-l_r^2CS_r(V_x)}{J_vV_x} - V_x\right)T_e + 1 & 0 & 0 \\ T_e & 0 & 1 & V_xT_e \\ 0 & 0 & T_e & 1 \end{bmatrix} \tag{C.7}$$

and

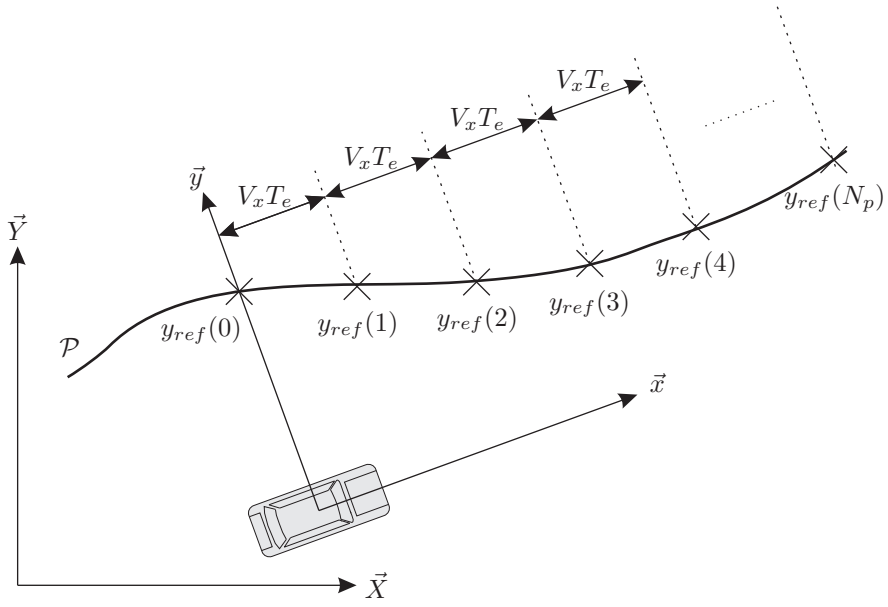



Figure C.2: Reference Points Definition

$$B_k = \begin{bmatrix} \frac{CS_f(V_x)}{M} T_e \\ \frac{l_f CS_f(V_x)}{J_v} T_e \\ 0 \\ 0 \end{bmatrix} \tag{C.8}$$

and $T_e$ the sample time.

The cost criterion defined in (C.1) requires $N_p$ reference points $y_{ref}$ from the path $\mathcal{P}$. First the point $y_{ref}(0)$, which is not used in the cost criterion is defined. This point is located at the intersection between the path and the $\vec{y}$ axis. It is used to determine the $N_p$ points $y_{ref}(k)$ of the path. Each point is then separated by $V_xT_e$ in the direction $\vec{x}$. Fig.C.2 presents the considered points.

## C.2 MPC Controller Tuning

This *MPC* Controller has been tested on *MatLab/Simulink* with a sample time of $20ms$. It has been implemented as a C-coded *S-Function*, using the QP solver routine available in *Matlab*, based on the publicly available Dantzig-Wolfe's algorithm [MathWorks, 2005]. The prediction horizon and the control horizon have the same length equal to 10 samples. To approach real testing conditions, a constraint is used to limit the maximum steering wheel angle (the maximum hand wheel angle is $\pm 450°$ thus involves a maximum steering wheel angle of $\pm 26°$) in order to mechanically protect the *Vehicle* steering column.

The tuning of the control solution has shown that the weightings $Q$ and $R$ are of the highest importance. On the one hand, a low value of $R$ generates an aggressive control insofar as the resulting angles and rotational speeds are high. This case is interesting when a quick reaction is required or when an important input is required to let the *Vehicle* move. However, it has been noticed that tracking oscillations may appear in such conditions. On the other hand, an

important value of $R$ allows a smoother control. Here, angles and rotational speeds are low. Such a setting is preferred when a small input leads to a strong reaction of the system. If such a value of $R$ helps to get a satisfying behavior of the *Vehicle*, the control is applied very late. Indeed, the optimization tries to limit the control without considering the path error. The weighting factors $Q$ and $R$ are here fixed to $Q = 1$ and $R = 2$ to give slightly more weight to the minimization of the control to be sent to the *Vehicle* steering wheel. This choice has been done as the Lateral Controller tries to match a constrained trajectory which has been designed to integrate some of the constraints which are usually considered during the Controller synthesis. In the previous section, it has been shown that constrained trajectories formulated as an optimization problem and based on the Parametric Cubic Spline model, provide curvature continuity. This property implies the trajectories to be smooth, so to present small variations of its curvature along the path. This is even more true considering the $WEM$ trajectory. The control required to cover the trajectory may consequently be reduced compared to an unconstrained trajectory, so requires less weight in the controller cost function. This is proved by the results of the following section.

The $LPV$ model of equation (C.5) is transformed into an $LTI$ model at each sample time by considering the *Vehicle* speed constant on the prediction horizon. This assumption is valid, due to the sample time that has been chosen ($20ms$). With a prediction of 10 samples and a sample time of $20ms$, it is considered that the longitudinal speed $V_x$ is constant for a maximum period of $0.2s$. This assumption helps to obtain a constrained $QP$ problem.

# Appendix D

# Speed Limit Assistant Precisions

## D.1 Focal Element Table

This table has been modified since [Bradai, 2007] with removals (in red) and additions (in green).

Table D.1: Detected Speed and Associated Focal Elements

| Navigation Speed ($km.h^{-1}$) | Number of Focal Elements | Focal Elements ($km.h^{-1}$) |
|---|---|---|
| 5 | 1 | 50 |
| 10 | 1 | 50 |
| 20 | 1 | 50 |
| 30 | 1 | 50 |
| 45 | 2 | 30, 50 |
| 50 | 7 | 30, 70, 100, 110, 120, 130, *unlimited* |
| 60 | 0 | *none* |
| 70 | 2 | 50, 80, 90 |
| 80 | 4 | 50, 60, 70, 90 |
| 90 | 1 | 50, 70 |
| 100 | 3 | 50, 70, *unlimited* |
| 110 | 3 | 50, 70, 90 |
| 120 | 3 | 50, 70, *unlimited* |
| 130 | 4 | 50, 70, 90, 110 |
| *unlimited* | 0 | *none* |

## D.2 Speed Limit Assistant Architecture

The fusion architecture of the developed Speed Limit Assistant is in Fig.D.1. Some important improvements could be noticed:

- The first enhancement concerns the fairness of the fusion. In the new scheme the information is gathered from both sensors in all cases. In fact, if a sensor is not giving information, its corresponding speed is automatically set to *undefined* with the largest belief (1.0).
- This approach for sensor information gathering directly implies the Speed Limit Assistant to go over the multi-sensor fusion, whenever the sensors are not giving information. The fusion is consequently placed as a top priority task. In addition, the fusion is not checking for a focal element which is similar to the vision speed. If sensors disagree, the fusion is

still processed and obviously involves the generation of a conflict mass. Consequently, the conflict management process has been added.

- The behavior of the Speed Limit Assistant after the multi-sensor fusion has also been modified. The decision is now corresponding to the determination of a pair of speeds: one based on the information directly obtained from the multi-sensor combination and the other one which results from the selection of a speed after the conflict redistribution. This has been implemented regarding to the elements presented in the previous section.

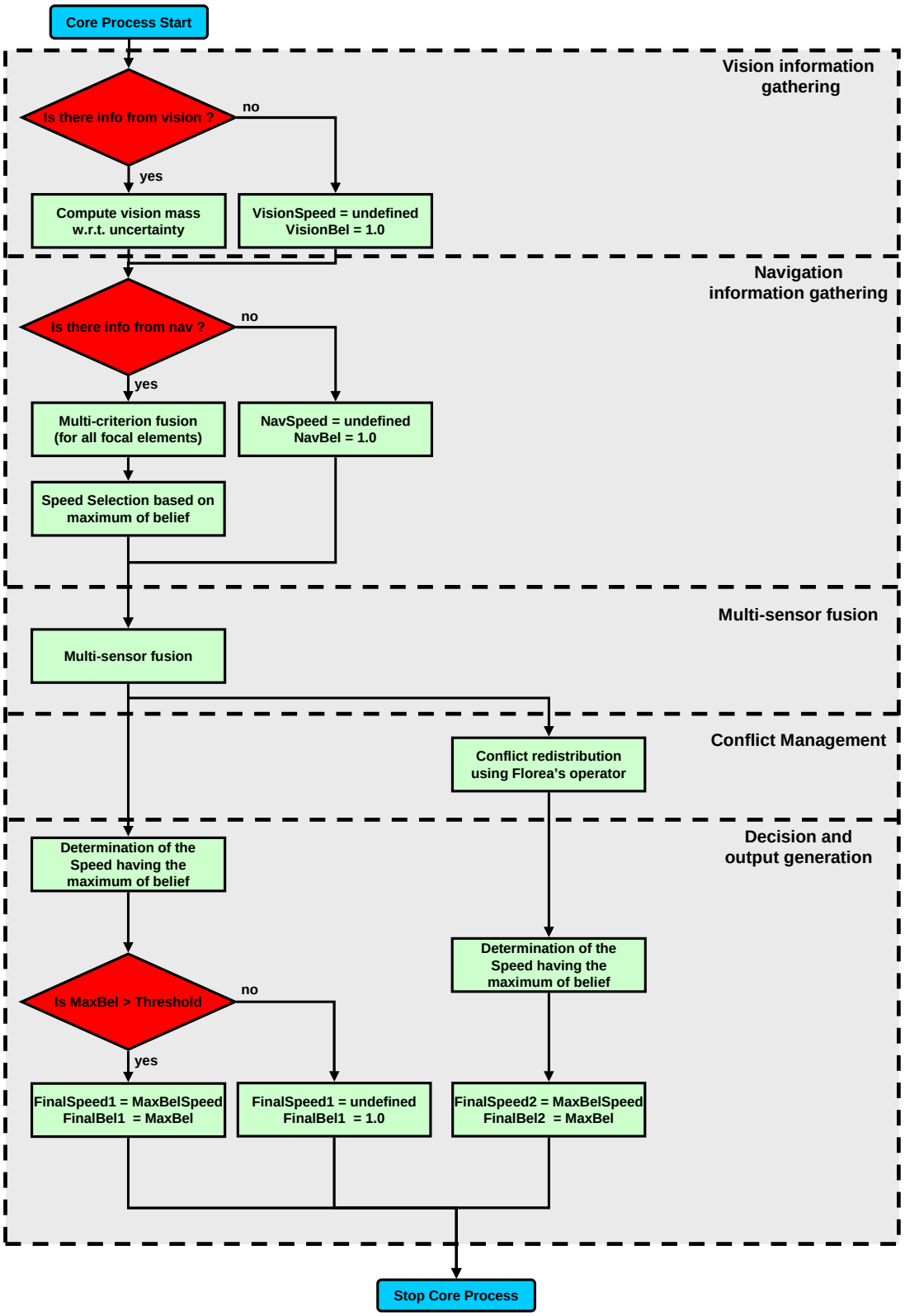



Figure D.1: SLA Fusion Architecture

# Bibliography


[ULT, 2010] (2010). http://www.ultralingua.com/. 4

[Altafini, 1999] Altafini, C. (1999). Curve negotiating using polar polynomials for nonholonomic vehicles. In *IFAC World Congress*, Beijing, China. 11

[Appriou, 2001] Appriou, A. (2001). Situation assesment based on spatially ambiguous multisensor measurment. *International Journal of Intelligent Systems*, 16:1135–1166. 73

[Bargeton, 2009] Bargeton, A. (2009). *Fusion multi-sources pour l'interprétation de l'environment routier*. PhD thesis, Ecole Nationale Supérieure des Mines de Paris. 37, 77

[Beji and Bestaoui, 2001] Beji, L. and Bestaoui, Y. (2001). An adaptive control method of automated vehicles with integrated longitudinal and lateral dynamics in road following. In *Second Workshop on Robot Motion and Control*. 34

[Bellman, 1957] Bellman, R. (1957). *Dynamic programming*. Princeton University Press. 5

[Bevan et al., 2010] Bevan, G., Gollee, H., and O'Reilly, J. (2010). Trajectory generation for road vehicle obstacle avoidance using convex optimization. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*. (In press). 15

[Biannic, 1996] Biannic, J. (1996). *Commande robuste des systèmes à paramètres variables: Applications à l'aéronautique*. PhD thesis, Centre d'Étude et de Recherche de Toulouse. 4

[Birouche et al., 2009] Birouche, A., Basset, M., Daniel, J., and Lauffenburger, J. (2009). Trajectory generation combined with tracking control: a new approach. In *European Symposium of Vascular Biomaterials*, Strasbourg, France. xi, 2, 3, 4

[Bonnifait et al., 2008] Bonnifait, P., Jabbour, M., and Cherfaoui, V. (2008). Autonomous navigation in urban areas using gis-managed information. *International Journal of Vehicle Autonomous Systems (IJVAS)*, 6:84–103. 30

[Boor, 1978] Boor, C. D. (1978). A practical guide to spline. *New York: Springer-Verlag*. 11, 13, 46

[Boyd and Vandenberghe, 2004] Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press. 34, 61, 64

[Bradai, 2007] Bradai, B. (2007). *Optimisation des Lois de Commande d'Eclairage Automobile par Fusion de Données*. PhD thesis, Université de Haute Alsace. xi, 18, 24, 35, 36, 39, 70, 87, 88, 95, 98, 127, 149, 167

[Bradai et al., 2007] Bradai, B., Herbin, A., Lauffenburger, J., and Basset, M. (2007). Predictive navigation-based virtual sensor for enhanced lighting. In *International Symposium on Automotive Lighting (ISAL)*. 26

[CAMP, 2004] CAMP (2004). *Enhanced digital mapping project final report*. United States Department of Transportation, Washington D.C. 24, 28

[Caron et al., 2006] Caron, F., Duflos, E., Pomorski, D., and Vanheeghe, P. (2006). Gps/imu data fusion using multisensor kalman filtering: introduction of contextual aspects. *Information Fusion*, 7:221–230. 31

[Carsten and Cacciabue, 2007] Carsten, O. and Cacciabue, P. (2007). *Modelling Driver Behaviour in Automotive Environments*, chapter From driver models to modelling the driver: what do we really need to know about the driver?, pages 105–120. Springer. 9

[Chan and Chung, 2003] Chan, A. and Chung, M. (2003). Indoor-outdoor air quality relationships in vehicle: effect of driving environment and ventilation modes. *Atmospheric Environment*, 37:3795–3808. 6

[Chapelon, 2008] Chapelon, J. (2008). La sécurité routière en france, bilan de l'année 2008: Les grandes donnés de l'accidentologie. Technical report, Observatoire national interministériel de la sécurité routière. 32, 35

[Chikhi, 2006] Chikhi, F. (2006). *Système Prédictif et Préventif d'Aide à la Conduite*. PhD thesis, Université de Versailles Saint Quentin en Yvelines. 35

[Daniel, 2007] Daniel, J. (2007). Contribution à la conception et au développement d'un système d'aide au guidage de véhicules professionnels. Master's thesis, Université de Haute Alsace. 31, 87

[Daniel et al., 2009a] Daniel, J., Pouly, G., Birouche, A., Lauffenburger, J.-P., and Basset, M. (2009a). Navigation-based speed profile generation for an open road speed assistant. In *IFAC Symposium on Control in Transportation Systems (CTS09)*. 26, 48

[Daniel et al., 2009b] Daniel, J., Truong, C., Lauffenburger, J.-P., and Basset, M. (2009b). Real-time trajectory generation for advanced driver assistance systems applications. In *International Forum On Strategic Technologies (IFOST09)*. 26

[Dantzig and Wolfe, 1960] Dantzig, G. and Wolfe, P. (1960). Decomposition principle for linear programs. *Operations Research*, 8:101–111. 64

[Delingette et al., 1991] Delingette, H., Hebert, M., and Ikeuchi, K. (1991). Trajectory generation with curvature constraint based on energy minimization. In *International Workshop on Intelligent Robots and Systems (IROS)*. 62

[Dempster, 1967] Dempster, A. (1967). Upper and lower probabilities induced by a multivalued mapping. *Annals of Mathematical Statistics*, 38:325–339. 18, 70

[Denoeux, 1995] Denoeux, T. (1995). A k-nearest neighbor classification rule based on dempster-shafer theory. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 25:804–813. 73

[Dorf and Bishop, 2008] Dorf, R. and Bishop, R. (2008). *Modern Control Systems*. xxiii, 2, 5

[Duan et al., 1999] Duan, Q., Djidjeli, K., Price, W., and Twizell, E. (1999). Weighted rational cubic spline interpolation and its application. *Journal of computational and applied mathematics*, 117:121–135. 14

[Dubins, 1957] Dubins, L. (1957). On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79:497–516. 10

[Dubois and Prade, 1988] Dubois, D. and Prade, H. (1988). Representation and combination of uncertainty with belief functions and possibility measures. *Computational Intelligence*, 4:244–264. 75

[Durekovic et al., 2007] Durekovic, S., Sixt, J., Dreher, S., and Becker, W. (2007). Adasrp 2006 user's manual. Technical report, Navigation Technologies. 29

[Egerstedt et al., 2001] Egerstedt, M., Hu, X., and Stotsky, A. (2001). Control of mobile platforms using a virtual vehicle approach. *IEEE Transaction on Automatic Control*, 46:1777–1782. 34

[Egerstedt and Martin, 2001] Egerstedt, M. and Martin, C. (2001). Optimal trajectory planning and smoothing splines. *Automatica*, 37:1057–1064. 14

[Ehmanns and Hochstadter, 2000] Ehmanns, D. and Hochstadter, A. (2000). Driver-model of lane change maneuvers. In *World Congress on Intelligent Transportation Systems*. 8

[Fang et al., 2005] Fang, H., Lenain, R., Thuilot, B., and Martinet, P. (2005). Trajectory tracking control of farm vehicles in presence of sliding. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005)*, pages 58–63. 34

[Feng et al., 2009] Feng, Y., Teng, T., and Tan, A. (2009). Modelling situation awareness for context-aware decision support. *Experts Systems with Applications*, 36:455–463. 24

[Floater, 2008] Floater, M. (2008). On the deviation of a parametric cubic spline interpolant from its data polygon. *Computer Aided Geometric Design*, 25(3):148–156. 13, 47

[Florea et al., 2006] Florea, M., Dezert, J., Valin, P., Smarandache, F., and Jousselme, A. (2006). Adaptative combination rule and proportional conflict redistribution rule for information fusion. In *COgnitive systems with Interactive Sensors (COGIS)*. 75, 81, 94

[Fraichard and Scheuer, 2004] Fraichard, T. and Scheuer, A. (2004). From reeds and shepp's to continuous-curvature paths. *IEEE Transaction on Robotics and Automation*, 20:1025–1035. 4, 9, 11, 54, 64

[Gavrila, 1999] Gavrila, D. (1999). Traffic sign recognition revisited. In *DAGM Symposium fur Musterekennung*. 77

[Glaser et al., 2002] Glaser, S., Mammar, S., and Sainte-Marie, J. (2002). Lateral driving assistance using embedded driver-vehicle-road model. In *Conference on Engineering Systems Design and Analysis*, Istanbul, Turkey. 8

[Glaser et al., 2007] Glaser, S., Nouvelière, L., and Lusetti, B. (2007). Speed limitation based on advanced curve warning system. In *Intelligent Vehicles Symposium (IV)*, Istanbul, Turkey. 32, 50

[Gómez-Bravo et al., 2008] Gómez-Bravo, F., Cuesta, F., Ollero, A., and Viguria, A. (2008). Continuous curvature path generation based on ß-spline curves for parking manoeuvres. *Robotic Autonomous Systems*, 56(4):360–372. 11

[Graettinger and Krogh, 1989] Graettinger, T. and Krogh, B. H. (1989). Evaluation and time-scaling of trajectories for wheeled mobile robots. *Journal of dynamic systems, measurement, and control*, vol. 111:222–231. 54

[Gruyer, 1999] Gruyer, D. (1999). *Etude et traitement de données imparfaites pour le suivi mluti-objet: application aux situations routières.* PhD thesis, Université de Technologie de Compiègne. 71, 72

[Gruyer et al., 2005] Gruyer, D., Rakotonirainy, A., and Vrignon, J. (2005). Advancement in advanced driving assistance systems tools: Integrating vehicle dynamics, environmental perception and drivers' behaviours to assess vigilance. In *Intelligent Vehicles & Road Infrastructure Conference*, Victoria, Australia. 8

[Hayes et al., 2003] Hayes, I., Jackson, M., and Jones, C. (2003). *Lecture Notes in Computer Science*, chapter Determining the Specification of a Control System from That of Its Environment, pages 154–169. 2, 6

[Hellström et al., 2006] Hellström, T., Johansson, T., and Ringdahl, O. (2006). *Development of an Autonomous Forest Machine for Path Tracking*, pages 603–614. 34

[Howard and Kelly, 2007] Howard, T. and Kelly, A. (2007). Optimal rough terrain trajectory generation for wheeled mobile robots. *International Journal of Robotics Research*, 26(1):141–166. 15

[ISO, 1997] ISO (1997). Mechanical vibration and shock evaluation of human exposure to whole body vibration - part1: General requirements. ISO 2631-1. 10, 50

[Jabbour et al., 2008] Jabbour, M., Bonnifait, P., and Cherfaoui, V. (2008). Map-matching integrity using multi-hypothesis road tracking. *Journal of Intelligent Transportation Systems*, 12:189–201. 31

[Jaynes, 2003] Jaynes, E. (2003). *Probability theory: the logic of science.* Cambridge. 18

[Klein, 2004] Klein, L. (2004). *Sensor and Data Fusion: A Tool for Information Assessment and Decision Making.* SPIE PRESS. 20

[Kojima et al., 2008] Kojima, Y., Kidono, K., Takiguchi, A., Amano, Y., and Hashizume, T. (2008). Precise ego-localization by integration of gps and sensory-based odometry. In *International Symposium on Advanced Vehicle Control (AVEC).* 31

[Kuncheva et al., 2003] Kuncheva, L., Whittaker, C., and Shipp, C. (2003). Limits on the majority vote accuracy in classifier fusion. *Patern Analysis and Applications*, 3:245–258. 18

[Lam and Suen, 1997] Lam, L. and Suen, C. (1997). Application of majority voting to pattern recognition: An analysis of its behavior and performance. *IEEE Transaction on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 27:553–568. 18

[Langley, 1999] Langley, R. (1999). Dilution of precision. *GPS World*, 10:52–59. 86

[Lauffenburger, 2002] Lauffenburger, J. (2002). *Contribution à la surveillance temps-réel du système-conducteur - véhicule - environnement : élaboration d'un système intelligent d'aide à la conduite*. PhD thesis, Université de Haute Alsace. 8

[Lauffenburger et al., 2003] Lauffenburger, J., Basset, M., Coffin, F., and Gissinger, G. (2003). Driver-aid system using path-planning for lateral vehicle control. *Control Engineering Practice*, 11:217–231. 11, 14

[Lauffenburger et al., 2008] Lauffenburger, J., Bradai, B., Basset, M., and Nashashibi, F. (2008). Navigation and speed signs recognition fusion for enhanced vehicle location. In *IFAC World Congress (IFAC WC)*, Seoul, South Korea. 78, 80, 81

[Laurence, 1998] Laurence, P. (1998). *Modélisation de systèmes complexes, stables et pseudostables. Application à l'étude du comportement en régime établi des véhicules routiers*. PhD thesis, Université de Haute Alsace. 8

[Lefevre et al., 2000] Lefevre, E., Colot, O., Vannoorenberghe, and De Brucq, D. (2000). A generic framework for resolving the conflict in the combination of belief structures. In *International Conference on Information Fusion (FUSION)*. 75

[Lefevre et al., 1999] Lefevre, E., Vannoorenberghe, P., and Colot, O. (1999). Using information criteria in dempster-shafer's basic belief assignment. In *International Conference on Fuzzy Systems*. 73

[Lefort-Piat and Gissinger, 2002] Lefort-Piat, N. and Gissinger, G. (2002). *La voiture intelligente*. Hermès. 9

[Ljung, 1999] Ljung, L. (1999). *System Identification:Theory for the User*. 4

[Maciejowski, 2000] Maciejowski, J. (2000). *Predictive control: with constraints*. Prentice Hall. 66

[Manca, 2006] Manca, N. (2006). Desing of an adaptive cruise controller via mpc/pwa-based methods. Master's thesis, Universita Degli Studi di Cagliari. 32

[Marin, 1984] Marin, S. (1984). An approach to data parametrization in parametric cubic spline interpolation problems. *Journal of Approximation Theory*, 41:64–86. 11

[Martin, 2005] Martin, A. (2005). La fusion d'informations. Polycopié de cours. xi, 17, 19

[MathWorks, 2005] MathWorks (2005). Model predictive control toolbox. technical report. Technical report. 165

[Meguro et al., 2008] Meguro, J., Murata, T., Takiguchi, J., Amano, Y., and Hashizume, T. (2008). Gps accuracy improvement by satellite selection using omnidirectional infrared camera. In *Intelligent Robots and Systems (IROS)*, pages 1804–1810. 31

[Messac and Sivanandan, 1997] Messac, A. and Sivanandan, A. (1997). A new family of convex splines for data interpolation. *Computer Aided Geometric Design*, 15:39–59. 11

[Minoiu Enache et al., 2009] Minoiu Enache, N., Netto, M., Mammar, S., and Lusetti, B. (2009). Driver steering assistance for lane departure avoidance. *Control Engineering Practice*, 17:642–651. 45

[Mourllion, 2006] Mourllion, B. (2006). *Extention d'un système de perception embarqué par communication, application à la diminution du risque routier*. PhD thesis, Université de Paris-Sud XI, Faculté des sciences d'Orsay. xii, 79, 80, 90

[Nagy and Kelly, 2001] Nagy, B. and Kelly, A. (2001). Trajectory generation for car-like robots using cubic curvature polynomials. In *Field and Service Robots (FSR)*. 11

[Najjar, 2003] Najjar, M. E. B. E. (2003). *Localisation dynamique d'un véhicule sur une carte routière numérique pour l'assistance à la conduite*. PhD thesis, Université de Technologie Compiègne. xi, 30, 31

[Naranjo et al., 2003] Naranjo, J., Gonzalez, C., Reviejo, J., Garcia, R., and De Pedro, T. (2003). Adaptive fuzzy control for inter-vehicle gap keeping. *Intelligent Transportataions Systems*, 4:132–142. 32, 34

[Naus et al., 2008] Naus, G., Ploeg, J., Van de Molengraft, R., and Steinbuch, M. (2008). Explicit mpc design and performance-based tuning of an adaptive cruise control stop-&-go. In *Intelligent Vehicles Symposium (IV)*, Eindhoven, Netherlands. 32

[Nehmzowa and Walker, 2005] Nehmzowa, U. and Walker, K. (2005). Quantitative description of robot-environment interaction using chaos theory. *Robotics and Autonomous Systems*, 53:177–193. 6

[Nelson, 1989] Nelson, W. (1989). Continuous-curvature paths for autonomous vehicles. In *International Conference on Robotics and Automation (ICRA)*. 11, 12

[Nienhüser et al., 2009] Nienhüser, D., Gumpp, T., and Zöllner, J. (2009). A situation context aware dempster-shafer fusion of digital maps and a road sign recognition system. In *Intelligent Vehicles Symposium (IV)*, Xi'an, China. 78

[Noyer et al., 2008] Noyer, U., Schomerus, J., Mosebach, H., Gacnik, J., Loper, C., and Lemmer, K. (2008). Generating high precision maps for advanced guidance support. In *Intelligent Vehicles Symposium (IV)*, pages 871–876. 28

[Nunes and Conde Bento, 2007] Nunes, U. and Conde Bento, L. (2007). Data fusion and path-following controllers comparison for autonomous vehicles. *Nonlinear Dynamics*, 49:445—462. 34

[Orfila, 2009] Orfila, O. (2009). *Infuence de l'infrastructure routière sur l'occurrence des pertes de contrôle de véhicules légers en virage : modélisation et validation sur site expérimental*. PhD thesis, Université d'Evry-Val d'Essonne. 15

[Peden et al., 2004] Peden, M., Scurfield, R., Sleet, D., Mohan, D., Hyder, A., Jarawan, E., and Mathers, C. (2004). *World report on road traffic injury prevention*. World Health Organization. 24

[Peng et al., 2007] Peng, J., Wang, Y., and Sun, W. (2007). Trajectory-tracking control for mobile robot using recurrent fuzzy cerebellar model. *Neural Information Processing - Letters and reviews*, 11:15–23. 34

[Pinchard et al., 1996] Pinchard, O., Liegeois, A., and Pougnet, F. (1996). Generalized polar polynomials for vehicle path generation with dynamic constraints. In *International Conference on Robotics and Automation (ICRA)*, Mineapolis, USA. 11, 14

[Pollock, 1999] Pollock, D. (1999). Smoothing with cubic splines. In *Handbook of Time Series Analysis, Signal Processing, and Dynamics*, pages 293 – 322. Academic Press, London. 63

[Pontryagine et al., 1962] Pontryagine, L., Boltiansky, V., Gamkrelidze, R., and Michtchenko, E. (1962). *The mathematical theory of optimal process.* Editions de Moscou. 5

[Pouly, 2009] Pouly, G. (2009). *Analysis and synthesis of advanced control laws for vehicle ground guidance.* PhD thesis, Université de Haute-Alsace. 4, 65, 111, 117

[Priez, 2000] Priez, A. (2000). Attention chercheurs à bord. *revue R&D (Renault), La route de l'innovation*, 15:15–19. 24

[Puthon et al., 2010] Puthon, A., Nashashibi, F., and Bradai, B. (2010). Improvement of multi-sensor fusion in speed limit determination by quantifying navigation reliability. In *International Conference on Intelligent Transportation Systems (ITSC)*, Madeira, Portugal. 78, 80, 81, 95

[Putney, 2006] Putney, J. (2006). Reactive navigation of an autonomous ground vehicle using dynamic expanding zones. Master's thesis, faculty of the Virginia Polytechnic Institute. 34

[Rimenez et al., 2008] Rimenez, F., Aparicio, F., and Paez, J. (2008). Evaluation of in-vehicle dynamic speed assistance in spain: algorithm and driver behaviour. *IET Intelligent Transportation Systems*, 2:120–131. 26

[Rombaut, 1998] Rombaut, M. (1998). Decision in multi-obstacle matching process using dempster-shafer's theory. In *Advances in Vehicle Control and Safety (AVCS).* xii, 71, 72, 73

[Rombaut and Zhu, 2002] Rombaut, M. and Zhu, Y. (2002). Study on dempster-shafer theory for image segmentation applications. *Image and Vision Computing*, 20:15–23. 73

[Rominger and Martin, 2010] Rominger, C. and Martin, A. (2010). Recalage et fusion d'images sonar multivues: utilisation du conflit. In *Atelier Fouille de données complexes - Complexité liée aux données multiples, Extraction et Gestion des Connaissances*, Hammamet, Tunisie. 76

[Royère, 2002] Royère, C. (2002). *Contribution à la résolution du conflit dans le cadre de la théorie de l'évidence : Application à la perception et à la localisation de véhicules intelligents.* PhD thesis, Université de Technologie de Compiègne. 71, 72, 79, 90, 92

[Schmidt and Heß, 1988] Schmidt, J. and Heß, W. (1988). Positivity of cubic polynomials on intervals and positive spline interpolation. *BIT Numerical Mathematics*, 28(2):340–352. 57

[Shafer, 1976] Shafer, G. (1976). *A mathematical theory of evidence.* Princeton University Press. 17, 18, 38, 70, 71, 72, 74, 75

[Smarandache and Dezert, 2009] Smarandache, F. and Dezert, J. (2009). *Advances and Applications of DSmT for Information Fusion. Collected Works.* American Research Press. 17, 18

[Smets, 1990] Smets, P. (1990). Constructing the pignistic probability function in a context of uncertainty. *Uncertainty in Artificial Intelligence*, 5:29–39. 76

[Smets, 1993] Smets, P. (1993). The disjunctive rule of combination and the generalized bayesian theorem. *International Journal of Approximate Reasoning*, 9:1–35. 75

[Smets and Kennes, 1994] Smets, P. and Kennes, R. (1994). The transferable belief model. *Artificial Intelligence*, 66:191–234. 18, 71

[Solea and Nunes, 2006] Solea, R. and Nunes, U. (2006). Trajectory planning with velocity planner for fully-automated passenger vehicles. In *Intelligent Transportation Systems Conference (ITSC)*, Toronto, Canada. 10

[Steinberg et al., 1999] Steinberg, A., Bowman, C., and White, F. (1999). Revision of the jdl data fusion model. In *SPIE Conference. Sensor Fusion: Architectures, Algorithms and Applications III*. 20

[Wald, 1999] Wald, L. (1999). Definition and terms of refence in data fusion. *International Archives of Photogrammetry and Remote Sensing*, 32-7-4-3-W6. 17

[Walter and Pronzato, 1997] Walter, E. and Pronzato, L. (1997). *Identification of Parametric Models from Experimental Data*. Springer. 3

[Waltz and Llinas, 1990] Waltz, E. and Llinas, J. (1990). *Multisensor data fusion*. Artech House. 17

[Wit, 2000] Wit, J. (2000). *Vector pursuit path tracking for autonomous ground vehicles*. PhD thesis, University of Florida. 34

[Yager, 1987] Yager, R. (1987). On the dempster-shafer framework and new combination rules. *Information Sciences*, 41:93–137. 72, 75

[Ye and Qu, 1999] Ye, J. and Qu, R. (1999). Fairing of parametric cubic splines. *Mathematical And Computer Modelling*, 30:121–131. 62

[Zadeh, 1965] Zadeh, L. (1965). Fuzzy sets. *Information and Control*, 8:338–353. 18

[Zadeh, 1979] Zadeh, L. (1979). On the validity of dempster's rule of combination. Memo M 79/24, Univ. of California, Berkeley, 1979. 74

# Abstract

Since the origin of the automotive at the end of the $19^{th}$ century, the traffic flow is subject to a constant increase and, unfortunately, involves a constant augmentation of road accidents. Research studies such as the one performed by the World Health Organization, show alarming results about the number of injuries and fatalities due to these accidents. To reduce these figures, a solution lies in the development of Advanced Driver Assistance Systems ($ADAS$) which purpose is to help the *Driver* in his driving task. This research topic has been shown to be very dynamic and productive during the last decades. Indeed, several systems such as Anti-lock Braking System ($ABS$), Electronic Stability Program ($ESP$), Adaptive Cruise Control ($ACC$), Parking Maneuver Assistant ($PMA$), Dynamic Bending Light ($DBL$), etc. are yet market available and their benefits are now recognized by most of the drivers.

This first generation of $ADAS$ are usually designed to perform a specific task in the *Controller/Vehicle/Environment* framework and thus requires only microscopic information, so requires sensors which are only giving local information about an element of the *Vehicle* or of its *Environment*. On the opposite, the next $ADAS$ generation will have to consider more aspects, i.e. information and constraints about of the *Vehicle* and its *Environment*. Indeed, as they are designed to perform more complex tasks, they need a global view about the road context and the *Vehicle* configuration. For example, longitudinal control requires information about the road configuration (straight line, bend, etc.) and about the eventual presence of other road users (vehicles, trucks, etc.) to determine the best reference speed. In addition, it requires several *Vehicle* information to determine its current configuration which allows to decide whether the *Vehicle* has to slow down or to accelerate. Regarding this example, it emerges that the relationships between the *Reference*, the *Controller*, the *Vehicle*, and the *Environment* have to be studied and refined.

The present PhD deals with both of these aspects: the constraints management and the information management between the *Vehicle/Environment* and the *Controller/Reference*:

- Constraints involved by the *Vehicle* and its *Environment* are usually considered in the *Controller* synthesis step so that the latter could become complex or even impossible to obtain considering restrictive hypotheses. To reduce the complexity of the *Controller*, the first contribution consists in the distribution of the constraints over the *Controller* **and/or** the *Reference*. For the next $ADAS$ generation, the *Reference* can be represented by the generation of unconstrained or constrained trajectories. Indeed, trajectories may provide a lot of information about the road context which can then be used for numerous applications. The contribution here lies in the definition of the trajectory generation as an optimization problem: considering constraints related to the *Vehicle*, to the *Driver*, and its *Environment* such as the road lane geometry given by a navigation system's *Digital Map Database*, the *Vehicle* minimal braking radius, the *Driver* acceleration limits, etc., a quadratic optimization looks for the optimal Parametric Cubic Spline coefficients while minimizing a predefined cost criterion. The comparison of different cost criteria allows to show that the minimization of the trajectory curvature is particularly interesting in the current power saving context. The validation of this approach is obtained through the comparison to an unconstrained trajectory generation over the longitudinal and the lateral

control of a car-like *Vehicle*.

- On the other hand, the management of the information is also of great importance as information may be inaccurate, redundant, erroneous, etc., thus may involve incoherent *Controller* behavior. The contribution, here, lies in the management of the information through Data Fusion. Indeed, the proposed fusion approach, based on the Evidence Theory, considers different levels of information, performing a multi-level fusion structure. The proposed fusion strategy is devoted to the combination of the information coming from a navigation and a vision system. One of the main contributions is to determine the navigation information through a multi-criterion fusion which considers each attribute of the *Digital Map Database* as an independent and specialized source. This multi-criterion fusion helps to detect and correct the navigation errors and inaccuracies in the *Vehicle* location. The benefits of the proposed solution are shown through its application to a Speed Limit Assistant defining the appropriate speed with respect to the contextual information provided by the navigation system and a speed sign recognition system.

**Keywords:** ADAS, Navigation systems, Trajectory generation, Cubic Splines, Quadratic optimization, Multi-level fusion, multi-criterion fusion, Evidence Theory.

## Résumé

Depuis l'invention de l'automobile à la fin du $19^{eme}$ siècle, la taille du parc ainsi que l'importance du traffic routier n'ont cessées d'augmenter. Ceci a malheureusement été suivi par l'augmentation constante du nombre d'accidents routiers. Un grand nombre d'études et notamment un rapport fourni par l'Organisation Mondiale de la Santé, a présenté un état alarmant du nombre de blessés et de décès liés aux accidents routiers. Afin de réduire ces chiffres, une solution réside dans le développement de systèmes d'aide à la conduite qui ont pour but d'assister le conducteur dans sa tâche de conduite. La recherche dans le domaine des aides à la conduite s'est montrée très dynamique et productive durant les vingt dernières années puisque des systèmes tels que l'antiblocage de sécurité ($ABS$), le programme de stabilité électronique ($ESP$), le régulateur de vitesse intelligent ($ACC$), l'assistant aux manœuvres de parking ($PMA$), les phares orientables ($DBL$), etc. sont maintenant commercialisés et acceptés par la majorité des conducteurs.

Cependant, si ces systèmes ont permis d'améliorer la sécurité des conducteurs, de nombreuses pistes sont encore à explorer. En effet, les systèmes d'aide à la conduite existants ont un comportement microscopique, en d'autres termes ils se focalisent uniquement sur la tâche qu'ils ont à effectuer. Partant du principe que la collaboration entre toutes ces aides à la conduite est plus efficace que leur utilisation en parallèle, une approache globale d'aide à la conduite devient nécéssaire. Ceci se traduit par la nécécité de développer une nouvelle génération d'aide à la conduite, prenant en compte d'avantage d'informations et de contraintes liées au véhicule, au conducteur et à son environnement. Par exemple, la régulation de vitesse requiert des informations liées à la composition de la route (lignes droites, virages, etc.) ainsi que des informations liées à la présence d'autre entités (automobiles, poids-lourds, etc.) afin de déterminer la vistesse de référence la plus sûre pour le véhicule. De plus, ce système requiert aussi des informations concernant la configuration actuelle du véhicule afin de déterminer si celui-ci doit accélérer ou freiner. De plus il ne faut pas oublier que l'importance de la consigne pour un un contrôleur, puisqu'elle représente la configuration à atteindre. Ceci étant, il apparait que les différentes interactions entre la *Référence*, le *Contrôleur*, le *Véhicule* et son *Environnement* doivent être prises en compte afin de fiabiliser les aides à la conduite.

Cette thèse s'inscrit au travers de ces deux aspects: la gestion des contraintes et le traitement des informations entre la paire *Véhicule/Environnement* et la paire *Contrôleur/Référence*:

- Premièrement, il est à rappeller que les contraintes liées au *Véhicule* et à son *Environnement* sont habituellement considérées lors de la synthèse du *Contrôleur* de telle manière que cette étape devienne très souvent complexe. Afin de la faciliter, la première contribution de cette thèse consiste en la répartition des contraintes sur le *Contrôleur* **et/ou** sur la *Référence*. En effet, la définition d'une *Référence* pré-contrainte permet de simplifier les démarches de synthèse du contrôleur. Pour la future génération d'aide à la conduite intégrant une approche plus globale, la *Référence* peut être représentée par la détermination de trajectoires contraintes ou non-contraintes, puisque celles-ci fournissent des informations variées sur le contexte routier pouvant être utilisées par de multiples applications. L'approche considérée ici consiste donc en la génération de trajectoire sous contraintes, formulée selon un problème d'optimisation. Suivant des contraintes liées au *Véhicule*, au *Conducteur* et à l'*Environnement* (géométrie de la voie de circulation, rayon de braquage minimal du véhicule, accélération limite supportée par le conducteur, etc.), une optimisation quadratique détermine les cœfficients optimaux d'un Spline Cubique Paramétrique en minimisant un critère de coût prédéfini. La mise en œuvre de différents critères de coût (distance, courbure, etc.) permet entre autre de montrer que la minimisation de la courbure de la trajectoire est particulièrement intéressante dans un contexte de réduction energétique. Cette méthode est validée au travers de sa comparaison avec des trajectoires non contraintes appliquées au contrôle longitudinal et latéral d'un véhicule.

- Deuxièmement, le traitement des informations multiples provenant de différentes sources et permettant de qualifier le contexte de conduite est aussi une étape importante puisque les informations peuvent être imprécises, incertaines, redondantes, fausses, etc., et peuvent de ce fait aboutir à un comportement abérrant de la part du *Contrôleur*. La contribution réside ici dans le traitement des informations au travers de la *Fusion de Données*. En effet, l'approche de fusion proposée ici, basée sur la théorie des croyances, considère plusieurs niveaux d'informations, et propose donc une approche de fusion multi-niveaux. Elle est notamment dédiée à la combinaision d'informations provenant d'un système de navigation et d'un dispositif de détection de panneaux de signalisation. Une des principales contributions consiste dans le traitement des informations de la navigation au travers d'une fusion multi-critères qui considère chaque source, caractérisant un critère, comme une source d'information spécialisée et indépendante. Cette étape permet de détecter les erreurs de la navigation tout en intégrant les incertitudes la caractérisant (positionnement, localisation, cartographie numérique). Les bénéfices de cette approche sont présentés au travers de son application à un assistant à la détérmination de vitesse limite ("Speed Limit Assisant") qui permet de définir la vitesse la plus adaptée au contexte routier.

**Mots-clés:** Aides à la conduite, Système de navigation, Génération de trajectoires, Splines cubiques, Optimisation quadratique, Fusion multi-niveaux, Fusion multi-critères, Théorie des Croyances.